PB84-135516

Generalization of Water Quality Criteria Using Chemical Models: Development of the REDEQL-UMD System of Computer Programs for Aqueous Equilibria. Part 2. Programmers Manual

Minnesota Univ.-Duluth

Prepared for

Environmental Research Lab.-Duluth, MN

Jan 84

U.S. Department of Commerce
National Technical Information Service
NTIS

PB84-135516

EPA-600/3-84-007b
January 1984

GENERALIZATION OF WATER QUALITY CRITERIA USING CHEMICAL MODELS -- Development of the REDEQL-UMD System of Computer Programs for Aqueous Equilibria

Part II

PROGRAMMERS MANUAL FOR REDEQL-UMD

A Detailed Description of the Program Including the Computational Algorithm, List of Variables, and Flow Diagrams

Donald K. Harriss, Sara E. Ingle, David K. Taylor, and Vincent R. Magnuson

Department of Chemistry
University of Minnesota-Duluth
Duluth, Minnesota 55812

Grant R804996

Project Officer

Gary E. Glass

Environmental Research Laboratory-Duluth
Duluth, Minnesota 55804

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

ENVIRONMENTAL RESEARCH LABORATORY-DULUTH
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
DULUTH, MINNESOTA 55804

TECHNICAL REPORT DATA

(Please read Instructions on the reverse before completing)

1. REPORT NO. EPA-600/3-84-007b	2.	3. RECIPIENT'S ACCESSION NO. PB84 135516
4. TITLE AND SUBTITLE GENERALIZATION OF WATER QUALITY CRITERIA USING CHEMICAL MODELS -- Development of the REDEQL-UMD System of Computer Programs for Aqueous Equilibria: Part II		5. REPORT DATE January 1984
		6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) D.K. Harris, S.E. Ingle, D.K. Taylor, and V.R. Magnuson		8. PERFORMING ORGANIZATION REPORT NO.
9. PERFORMING ORGANIZATION NAME AND ADDRESS University of Minnesota-Duluth Duluth, Minnesota 55812		10. PROGRAM ELEMENT NO.
		11. CONTRACT/GRANT NO. R804996
12. SPONSORING AGENCY NAME AND ADDRESS U.S. Environmental Protection Agency Environmental Research Laboratory-Duluth 6201 Congdon Boulevard Duluth, Minnesota 55804		13. TYPE OF REPORT AND PERIOD COVERED
		14. SPONSORING AGENCY CODE EPA/600/03

15. SUPPLEMENTARY NOTES

Part II. Programmers Manual for REDEQL-UMD, A Detailed Description of the Program Including the Computational Algorithm, List of Variables, and Flow Diagrams

16. ABSTRACT

This research program had the primary objective of developing site-specific water quality criteria for metals through correlating aquatic toxicity with activities of metal species. The project was not intended to carry out experimental studies but to make use of existing literature reports of metal toxicity to generate the correlations.

Inherent in these studies was the need for knowledge of the number of metal species present, and the relative amounts of each. To meet these needs a set of computer programs for finding equilibrium concentrations of metal complexes REDEQL-UMD, was developed, along with a stability constant data base and associated reference-footnote data bases.

Literature values of stability constants were adjusted to 25° C by application of thermodynamic principles.

Once speciation and toxicity data were available, a combination of factor analysis and multiple regression analysis was used to develop correlations. A detailed example involving toxicity of copper to Daphnia magna is included.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group

18. DISTRIBUTION STATEMENT RELEASE TO PUBLIC	19. SECURITY CLASS *(This Report)* UNCLASSIFIED	21. NO. OF PAGES 233
	20. SECURITY CLASS *(This page)* UNCLASSIFIED	22. PRICE

NOTICE

THIS DOCUMENT HAS BEEN REPRODUCED FROM THE BEST COPY FURNISHED US BY THE SPONSORING AGENCY. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE.

NOTICE

This document has been reviewed in accordance with U.S. Environmental Protection Agency policy and approved for publication. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

ABSTRACT

Con't from BDS

This programmers manual consists of three parts:

A. Definitions and Computational Algorithms for REDEQL-UMD in which the algorithms for all computational processes are described, with examples used when the procedure is complex.

B. Parameters and Global Variables for REDEQL-UMD in which the global variables are defined as to nature and type, and normal ranges are given as well as the names of subroutines in which they are used and the kind of use made of the variables in those subroutines.

C. Flow Diagrams for REDEQL-UMD in which flow diagrams for all subroutines are given in considerable detail as well as a general flow sheet for the program.

The manual will find greatest use by those desiring to make major modifications to REDEQL-UMD although it would be useful to a serious user wishing to obtain a more thorough understanding of the set of programs.

CONTENTS

Page

Abstract iv

Tables vi

Figures vi

1. Introduction 1
2. Definitions and Computational Algorithms for REDEQL-UMD . 2
3. Parameters and Global Variables for REDEQL-UMD 32
4. Flow Diagrams for REDEQL-UMD 46

TABLES

Number Page

1. Numbering system and types of complexes 5

FIGURES

Number Page

1. Calculational Flow for REDEQL-UMD 47
2. REDEQL-UMD Main Program 48
3. SUB1 50
4. SUB2 52
5. SUB3 54
6. ACTLOG 60
7. ADSORP 62
8. ALLOTC 65
9. ALLOTS 68
10. ANOUNC 70
11. BALANC 72
12. CHEKML 75
13. CHKSOL 77
14. CO3ALK 83
15. COLECT 86
16. CVTEST 88
17. DEFALT 90
18. DISSOL 92
19. GAM & GAMA 99
20. GAUSSL 101
21. INADS 103
22. INCONC 106

23. INMAT 110
24. INMIX 124
25. INMIXC 129
26. INRED 131
27. INVERT 137
28. IONCOR 142
29. IONICS 144
30. ITERTN 146
31. MIXOL2 148
32. MLINS1 150
33. MODIX 152
34. OPTSO 154
35. OUT1 156
36. OUT17 159
37. OUT139 163
38. OUT140 167
39. OUT141 169
40. OUTCNC 171
41. OUTCST 173
42. OUTIN 177
43. OUTJAC 181
44. OUTMC 183
45. OUTSAT 186
46. OUTSOL 188
47. PAGE 190
48. REDOX1 192

49. SCAN1 194
50. SECVAR 196
51. SKIP 199
52. SOLID2 201
53. SUBJAC 203
54. SUBMOL 206
55. SUBXC 209
56. SUBX 211
57. SUBZ 213
58. TESTS 217
59. TOTCC 219
60. ZERRIN & ZERROR 222

INTRODUCTION

The Users Guide to REDEQL-UMD is sufficiently detailed to satisfy the needs of the majority of users of this program. In addition, the program listings are liberally supplied with COMMENTS to assist in making minor changes when such are desired and when the necessary program changes are easily identified.

The set of programs are, however, sufficiently complex that most modifications would be extremely difficult with only the above resources. Another problem of the same magnitude may arise if an attempt is made to adapt the set of programs to a computer system which is substantially different, e.g., much smaller core memory or very different operating system.

To meet these needs, this Programmers Manual provides descriptions of the computational algorithms, a complete description of variable use, and flow sheets for the subroutines.

DEFINITIONS AND COMPUTATIONAL ALGORITHMS FOR REDEQL-UMD

Many of the definitions and computational procedures used in REDEQL-UMD are sufficiently complex that detailed descriptions and explicit examples are required.

In the following pages error and convergence are defined along with the modifications made necessary when solids exist or oxidation-reduction is allowed.

Introduction of solids or oxidation-reduction reduces the number of degrees of freedom, necessitating a separation of variables into principal variables and secondary variables. The definition of secondary variables and the procedures used in their selection, both for solids and for oxidation-reduction, are treated in detail. Changes in calculations involving secondary variables are discussed.

DEFINITIONS AND COMPUTATIONAL ALGORITHMS FOR REDEQL-UMD

I. INTRODUCTION

II. SOLIDS NOT CONSIDERED

  A. DEFINITION OF ERROR
  B. DEFINITION OF CONVERGENCE WITH RESPECT TO MASS
  C. DEFINITION OF Z MATRIX
  D. SOLUTION OF EQUATIONS FOR VARIATIONS IN FREE CONCENTRATIONS
  E. CALCULATION OF NEW FREE CONCENTRATIONS
  F. CALCULATION OF IONIC STRENGTH AND CORRECTION OF THERMODYNAMIC CONSTANTS
  G. CALCULATION OF CHARGE ON THE SOLUTION

III. SOLIDS POSSIBLE

  A. DETERMINATION OF SECONDARY VARIABLES AND STORAGE OF THEIR DEPENDENCE UPON PRINCIPAL VARIABLES
  B. MODIFICATION OF THE ERROR VECTOR AND THE Z MATRIX TO ACCOUNT FOR AMOUNTS IN SOLIDS
  C. CALCULATION OF CONCENTRATIONS OF SOLIDS AND DISSOLUTION OF UNSATURATED SOLIDS
  D. CHECKS FOR SUPERSATURATION OF NON-PRECIPITATED SOLIDS

IV. ION-ION REDOX ALLOWED

  A. DEFINITION OF SECONDARY VARIABLES
  B. REDUCING THE NUMBER OF DEPENDENT VARIABLES TO EQUAL THE NUMBER OF SOLIDS
  C. CALCULATION OF TOTAL HYDROGEN AND ELECTRONS

SECTION I. INTRODUCTION

The program considers M {up to 50 in the standard program} metal and ligand species, i, with total metal concentrations, M(i), and free concentrations, X(i) <*INMAT*>. The species are identified by sequential numbers with metals listed first and H+ always being 1. Hydroxide is always the Mth species {M=NOH} and electrons, if included, are also M {M=NEL}. These species may join together to form complexes {up to 750 are allowed by the standard program} where the formation constant of complex k is K(k) and its concentration is C(k). The formation constants and stoichiometries {v(i,k)= stoichiometry of component i in complex k} are stored in the thermodynamic data file. For each complex k,

$$C(k) = K(k) \prod_{i=1}^{M} X(i)^{v(i,k)} \qquad \text{<*SUBXC*>} \qquad (1)$$

Hydroxide is never a variable for computations. Complexes containing OH- are written using hydrogen ions with a negative stoichiometry.

Adsorption constants are dependent upon pH so that

$$K(k) = KA(k) \times 10^{s(pH - pH(ZPC))(Q(i)v(i,k) + v(H,k))} \qquad \text{<*ADSORP*>}$$

where KA(k) is the intrinsic adsorption constant, s is the scaling parameter often called alpha, ZPC is the zero point of charge, i is the adsorbing metal in the k-th complex, and Q(i) is its charge.

The possible types of complexes and the numbering system are shown in Table 1 on the following page. Most complexes contain a metal, a ligand, and possibly hydrogen, and thus have at most three non-zero values of v(i,k). These complexes are numbered from 1 to NCREG. Redox complexes contain electrons so that v(NEL,k) is non-zero in Eq.(1). Redox complexes are numbered from NCREG+1 to NCRED. Mixed complexes contain an additional metal or ligand ion and thus possibly have four non-zero v(i,k). In sequence, mixed complex numbering starts at NCRED+1 and goes to NCMIX. Adsorption complexes constitute the fourth type and they are numbered NCMIX+1 to NC. Adsorption complexes differ from regular complexes in that their Ks are not corrected for ionic strength but are corrected for changes in pH. These complexes do not contribute to ionic strength.

NOTE: Curly brackets contain information pertinent to the sentence. Angle brackets indicate the subroutine(s) in which the immediately preceding process occurs.

TABLE 1. NUMBERING SYSTEM AND TYPES OF COMPLEXES

Numbering	Species	Alternative Numbering
Metals and Ligands		
1	H^+	
.		
.	metal ions	
.		
.		
.	ligands	
.		
.		
.	adsorbing surfaces	
.		
.		
M	OH^- and e^-	
Complexes		
1		
.		
.	regular	
.		
NCREG		
NCREG+1		1
.		.
.	redox	.
.		.
NCRED		MCRED
NCRED+1		1
.		.
.	mixed	.
.		.
NCMIX		MCMIX
NCMIX+1		
.		
.		
.	adsorption	
.		
NC		

(continued)

TABLE 1. (continued)

Numbering	Species	Alternative Numbering
Solids		
1		
.		
.	regular	
.		
NSREG		
NSREG+1		1
.		.
.	redox	.
.		.
NS		MSRED
Mixed Solids		
1		
.		
.		
.		
NMIX		

All stoichiometries are expressed as v(i,k) for component i in complex k and u(i,j) for component i in solid j. Storage of these coefficients would necessitate an array of v(50,750) for complexes and an array of u(50,100) for solids {assuming the the standard maximum of 100 solids}. To decrease space requirements, complexes use five A(750) arrays which require, of course, only one-tenth the space of v(50,750). The arrays are:

metal in complex k	CMET(k)
ligand in complex k	CLIG(k)
stoichiometry of the metal	CM(k)
stoichiometry of the ligand	CL(k)
stoichiometry of H+	CHOH(k)

so the program uses

$$C(k) = K(k)X(CMET(k))^{CM(k)} X(CLIG(k))^{CL(k)} X(H)^{CHOH(k)} .$$

For the few mixed complexes {standard storage provides for 100} two additional A(100) arrays are used:

third ion in complex k	CLIG2(k - NCRED)
stoichiometry of the third ion	CL2 (k - NCRED) .

For redox complexes {standard storage provides for 20} one extra A(20) array is used:

stoichiometry of electrons in complex	NELC(k - NCREG) .

The alternative numberings shown in Table 1 are used to identify positions in the extra arrays for mixed and redox complexes. While the above definitions and numbering system show the variables used and the order in which the program performs the calculations, the general notation previously introduced will be used for brevity in this text.

The problem which the program must solve can be viewed at three levels: Solids Not Considered (no dependent variables); Solids Possible (one dependent variable for each solid); Ion-Ion Redox Allowed (an additional dependent variable for each redox).

SECTION II. SOLIDS NOT CONSIDERED

A. DEFINITION OF ERROR

When components can be present only as free ions or in complexes, the total amount of i in solution, T(i), is the sum of the free concentration plus the amount of i in each complex k:

$$T(i) = X(i) + \sum_{k=1}^{NC} v(i,k)C(k) \quad . \qquad \text{<*SUBY*>} \qquad (2)$$

The computation is an iterative process which begins with concentrations of complexes calculated from Eq.(1) using K´s corrected to the fixed or last calculated ionic strength <*IONCOR*> and the free concentrations of components, X(i). For the first iteration, X(i) is set equal to 0.01 M(i) <*INMAT*> and for other cases to X(i) of the last case <*TOTCC*>.

Two exceptions to calculation of T(i) by Eq.(2) are T(1) for total hydrogen {which is free hydrogen less free hydroxide} whose free concentration is equal to

$$X(1) - Kw/X(1),$$

and T(NEL) for electrons whose free concentration is 0.

The error in mass of component i is Y(i) where

$$Y(i) = T(i) - M(i) \quad . \qquad \text{<*SUBY*>} \qquad (3)$$

If pH is being calculated, a value for total H+ is calculated before iterations for each case are begun. This is evaluated using

$$-M(H) = -TOTH = \sum_{\substack{i=2 \\ i/OH-}}^{M} Q(i)M(i) \qquad \text{<*TOTCC*>}$$

where Q(i) is the charge on the i-th species and TOTH is defined as total hydrogen, i.e. total hydrogen is determined from electroneutrality.

When pE is variable, an initial concentration, M(NEL), of electrons in complexes and solids must be input by the user. It does not include free electrons {there aren´t any} and it may be positive or negative, e.g. if formation of a complex produces electrons, the complex contains a negative concentration of electrons. Y(H) and Y(NEL) are calculated only when pH and pE, respectively, are variable. M(NEL) is conservative, i.e. the electrons can be shifted around but neither lost nor gained; for each thing oxidized, something else must be reduced.

One other exception to the conservation of mass equations is encountered when carbonate alkalinity is used as input rather than total inorganic carbon. Total carbonate alkalinity is defined as

$$T(alk) = 2\ X(CO3) + \sum_{k=1}^{NC} C(k)[2v(CO3,k) - v(H,k)] + Kw/X(1) - X(1) \qquad <\!*CO3ALK*\!>$$

where the summation is only over carbonate containing complexes. Corresponding values of Y(alk) and Z {defined in Section II.C.} are defined. The equation shown is only for soluble carbonate species so conservation of total carbonate alkalinity cannot be used when carbonate-containing solids are present. Attempts to precipitate a carbonate-containing solid when carbonate alkalinity had been specified will generate error messages. The soluble carbonate alkalinity is calculated and printed for every case.

All total concentrations, M(i) and TOTH, are conservative for the calculation unless they are changed to achieve electroneutrality or unless ion-ion redox is considered.

B. DEFINITION OF CONVERGENCE WITH RESPECT TO MASS

When convergence with respect to mass has been achieved {the sum of the parts equals the whole}, $T(i) = M(i)$ for all species i or, in theory, $Y(i) = 0$. Convergence is considered to be complete when

$$|Y(i)|/M(i) < 0.0001 \qquad <\!*CVTEST*\!> \qquad (4)$$

for all i.

C. DEFINITION OF THE Z MATRIX

If $|Y(j)|$ is too large according to Eq.(4), X(i) need to be changed, thus changing T(j). The error Y(j) is related to the errors in free concentrations by the matrix Z

$$Y(j) = DT(j) = \sum_{i=1}^{M} [\partial T(j)/\partial X(i)]DX(i) = \sum_{i=1}^{M} Z(j,i)DX(i) \qquad j=1,M \qquad (5)$$

where DT(j) and DX(i) are the changes in T(j) and X(i). The elements of Z are found by substitution of Eq.(1) into Eq.(2) and derivation of the following equation:

$$Z(j,i) = [\partial T(j)/\partial X(i)] = [\partial X(j)/\partial X(i)] + \sum_{k=1}^{NC} v(j,k)K(k)[\partial/\partial X(i)]\prod_{m=1}^{M} X(m)^{v(m,k)}$$

$$= [\partial X(j)/\partial X(i)] + \sum_{k=1}^{NC} v(j,k)K(k)v(i,k)X(i)^{v(i,k)-1} \prod_{\substack{m=1\\ m/i}}^{M} X(m)^{v(m,k)}$$

$$= [\partial X(j)/\partial X(i)] + \sum_{k=1}^{NC} v(j,k)v(i,k)K(k)X(i)^{-1} \prod_{m=1}^{M} X(m)^{v(m,k)}$$

$$= [\partial X(j)/\partial X(i)] + \sum_{k=1}^{NC} v(j,k)v(i,k)C(k)X(i)^{-1} \qquad \text{<*SUBZ*>} \qquad (6)$$

Because of the inclusion of free hydroxide in total hydrogen, Z(1,1) is given by:

$$Z(1,1) = 1 + Kw/X(H)^2 + \sum_{k=1}^{NC} v(1,k)^2 C(k)/X(H) \ .$$

When pH is fixed, Y(1) does not have to converge to zero and the first row and column of the Z matrix are not used.

For electrons,

$$Z(NEL,NEL) = \sum_{k=1}^{NC} v(NEL,k)^2 C(k)/X(NEL) \qquad \text{<*SUBZ*>}$$

since there are no free electrons.

D. SOLUTION OF EQUATIONS FOR VARIATIONS IN FREE CONCENTRATIONS

When all components of Z have been calculated, the series of M equations described by Eq.(5) must be solved for the unknowns DX(i) <*GAUSSL*>. If pH is fixed there are only M-1 equations, from Y(2) to Y(M). M may be the last ligand, {=NOH-1}, the last adsorbing surface {=NOH-1}, or electrons {=NEL=NOH} if pE is variable. There is never an equation for OH-. The series of equations are better written in matrix notation as

$$\begin{vmatrix} Y(1) \\ Y(2) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ Y(M) \end{vmatrix} = \begin{vmatrix} Z(1,1) & Z(1,2) & \cdots\cdots & Z(1,M) \\ Z(2,1) & Z(2,2) & \cdots\cdots & Z(2,M) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ Z(M,1) & Z(M,2) & \cdots\cdots & Z(M,M) \end{vmatrix} \begin{vmatrix} DX(1) \\ DX(2) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ DX(M) \end{vmatrix} \qquad (5a)$$

Eq.(5a) is solved by Gaussian elimination with partial pivoting {Ref: V.N. Faddeeva (Translated from Russian by Curtis D. Benster), "Computational Methods of Linear Algebra", Dover Publications, Inc.(1959), pp. 65-72}. In this procedure, all elements below the diagonal are reduced to zero {Z(i,j) = 0 if i>j}. Since the order of the equations is irrelevant and multiples of any one equation may be added to any other equation without affecting the solution, it is not difficult to transform the equations to the desired form. The zeroing procedure is done one column at a time, from left to right. If, below the diagonal element, any element in the column is larger than the diagonal element, the equation containing the large element is exchanged with the equation containing the diagonal. Thus, the array rows and the Ys corresponding to these rows are interchanged. Multiples of the diagonal-containing equation then are subtracted from each lower equation so that all elements below the diagonal in this column become zero. The procedure then is repeated with the next column. The equations are easy to solve, from the bottom row to the top. The bottom equation contains only one variable, DX(M), whose value is readily determined. The next to last row contains variables DM(M-1) and DX(M), and again is readily solved since DX(M) is known. Similarly the M-2nd row is solved for DX(M-2), with DX(M) and DX(M-1) known.

E. CALCULATION OF NEW FREE CONCENTRATIONS

The DX(i) are negative corrections to the free concentrations X(i):

$$X(i) = X(i)old - DX(i) \qquad \text{<*ITERTN*>} \qquad (7)$$

unless

$$X(i)old - DX(i) < 0 \quad . \qquad (8)$$

Species cannot have negative free concentrations, so when relationship (8) is found to be true, the formula

$$X(i) = 0.1\ X(i)old \qquad \text{<*ITERTN*>} \qquad (9)$$

is used instead of Eq.(7).

If the pH is changed, all K(k)s for adsorption {k>NCMIX} are recalculated <*ADSORP*>. The C(k) {Eq.(1)} then are calculated from the new X(i)

<*SUBXC*>, the Y(i) and the Z(i,j) {Eqs.(2) and (3)} are evaluated <*SUBY*,*SUBZ*>, the Y(i) are compared to the M(i) for convergence with respect to mass <*CVTEST*> and, if for any i Eq.(4) is not satisfied, new corrections DX(i) are determined {Eq.(5)} <*GAUSSL*> and the procedure repeated.

F. CALCULATION OF IONIC STRENGTH

When Eq.(4) is true for all i the convergence with respect to mass is complete. If the ionic strength is allowed to vary, it is calculated according to

$$I = 1/2 \sum_{i=1}^{M} Q(i)^{2} X(i) + 1/2 \sum_{k=1}^{NCMIX} Q(k)^{2} C(k) \qquad \text{<*IONICS*>} \qquad (10)$$

where Q(i) and Q(k) are the charges on species i and complex k respectively. If

$$I - Iold < 0.005\ I \qquad (11)$$

the change in ionic strength is considered insignificant. If Eq.(11) does not hold, all constants K must be corrected for variations in activity with ionic strength <*IONCOR*> and the computations for convergence with respect to mass must be restarted.

A corrected constant K′ is related to the thermodynamic constant K by

$$K'(k) = K(k) \prod_{i=1}^{M} GAM(i)^{v(i,k)} / GAM(k)$$

where GAM(k) and GAM(i) are the activity coefficients of complex k and ion i, respectively.

Either of two equations for activity coefficients may be used. At 25^{0} C they are the Magnuson equation given by

$$-\log GAM(i) = 0.509\ Q(i)^{2} [I^{1/2} / (1 + 1.51\ I^{1/2}) - 0.08\ I] - 0.09\ I \quad ,$$

and the Davies equation

$$-\log GAM(i) = 0.509\ Q(i)^{2} [I^{1/2} / (1 + I^{1/2}) - 0.3\ I] \quad .$$

G. CALCULATION OF CHARGE ON THE SOLUTION

When both Eqs.(4) and (11) are satisfied, one final computation is made if pH is fixed. This is the calculation of the charge on the solution:

$$Q(total) = \sum_{i=1}^{M} Q(i)T(i) \quad . \qquad \text{<*BALANC*>} \qquad (12)$$

If

$$|Q(total)| < 0.001\ I \quad , \qquad \text{<*BALANC*>} \qquad (13)$$

electroneutrality is considered to have been achieved. If Eq.(13) is not true, either a warning message is printed or one or more M(i) {and resulting T(i)} are adjusted and iterations for convergence with respect to mass are begun again.

The results printed <*OUT1*> when Eqs.(4), (11), and (13) are satisfied are: free concentrations X(i), activities GAM(i)X(i), total concentrations T(i), concentrations of complexes C(k) <*OUT17*>, pH, and ionic strength. Other related parameters, such as [OH−] or pOH and K′s corrected to the final ionic strength, also may be printed.

SECTION III. SOLIDS POSSIBLE

A solid may be precipitated or a solution may be supersaturated or undersaturated with respect to a solid. If the user does not allow a solid to precipitate {i.e. permits supersaturation} the supersaturation has no effect on computation. This is effected by ignoring the formation constant, S(j), of solid j in all calculations. When all computations have been completed, the ion product of solid j,

$$\prod_{i=1}^{M} X(i)^{u(i,j)} \quad , \qquad <*DISSOL*>$$

{where u(i,j) is the stoichiometry of species i in solid j} is compared to S(j) to determine the degree of saturation.

If a solid precipitates, calculations are greatly affected. A solid may precipitate for either of two reasons: because the user has required that it precipitate before any calculations are made {an imposed solid}, or because a solid which is not expressly prohibited from precipitating is determined to be supersaturated after all mass balance equations for the system have been satisfied. The presence of a solid reduces the number of independent variables because, for a solution saturated with respect to solid j,

$$\prod_{i=1}^{M} X(i)^{u(i,j)} = S(j)^{-1} \quad . \qquad (14)$$

Any one of the X(i) can be written in terms of S(j) and the other X(i). The smallest X(i) is chosen <*SCAN1*> to depend on all others and it becomes a dependent {secondary} variable. If n is the secondary variable

$$\log X(n) = -\sum_{\substack{i=1 \\ i\neq n}}^{M} u(i,j)X(i)/u(n,j) - \log S(j)/u(n,j) \quad . \qquad (14a)$$

The program considers four types of solids:

Regular	M + L + H
Redox	M + L + H + e
Regular Mixed	M1 + M2 + L1 + L2 + H

Clays $M1 + M2 + M3 + M4 + Al^{3+} + SiO_3^{2-} + H$

All are equally well described by Eqs.(14) and (14a).

The effects of solids must be considered in four areas: Determination of a secondary variable for each solid and storage in the COEX matrix of the dependence of this variable upon principal variables; Inclusion in T(i) of the amounts of i in solids, the modification of Z(i,j) for amounts of j in solids, and variations of variables dependent upon X(i); Calculation of solid concentrations for each precipitated solid; Checks for supersaturation of non-precipitated solids.

A. DETERMINATION OF SECONDARY VARIABLES AND STORAGE OF THEIR DEPENDENCE UPON PRINCIPAL VARIABLES

The COEX matrix is generated to keep track of the dependency of each secondary variable upon the principal variables. Each element of the matrix, COEX(i,j), specifies the dependence of variable i upon variable j. If all of the variables are independent, the COEX matrix is a unit matrix. When there are dependent variables, one diagonal element will be equal to zero for each dependent variable. The row, n, which contains a zero diagonal element {COEX(n,n)=0} will be an equation for the dependent variable X(n) in terms of the independent variables X(i),

$$- \log X(n) = \sum_{i=1}^{M} COEX(n,i)[-\log X(i)] + COEX(1,n) \quad . \qquad (15)$$

From comparison of Eqs.(15) and (14a), one sees that for a dependent variable in a single solid j,

$$COEX(n,i) = -u(i,j)/u(n,j) \quad .$$

In general, an array element COEX(n,i) will be non-zero only if n=i and i is independent, or i is an independent variable and n is dependent upon it. The calculation of COEX elements is illustrated below for a series of four solids among seven components. A mixed solid would be treated in exactly the same manner as a regular solid except that it would contain more components and the dependent variable selected would depend upon more principal variables. Prior to solid formation, COEX is a unit matrix:

$$\begin{array}{c|ccccccc|}
 & H & M & N & O & F & G & L \\
\hline
H & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
M & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
N & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
O & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
F & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
G & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
L & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

If a first solid, $M_{a(1)}L_{b(1)}H_{c(1)}$, with formation constant S(1) precipitates,

$$S(1) = X(M)^{-a(1)} \; X(L)^{-b(1)} \; X(H)^{-c(1)}$$

$$\text{where } a(1) = u(M,1), \quad b(1) = u(L,1), \text{ and } \quad c(1) = u(H,1) \quad .$$

Since both M and L are independent variables {H is also independent but isn't listed since it never can be dependent}, either can be the dependent variable. In general, the program selects candidates for the dependent variable by calculating a dependence vector, D, for each variable j.

$$D(j) = a(1)COEX(M,j) + b(1)COEX(L,j) \qquad j=1,7 \qquad \text{<*SOLID2*>} \quad . \quad (16)$$

and

$$D(1) = D(1) + c(1)$$

Since there aren't any dependent variables yet and the matrix is a unit matrix,

$$D(M) = a(1), \quad D(L) = b(1), \quad D(H) = c(1), \text{ and } D(j) = 0 \quad j \;/\; M,L,H$$

If D(j) / 0 and j / 1, j may be a dependent variable. LogK for the formation of the solid is modified if any of the components are dependent variables

$$\begin{aligned} \log K &= \log S(1) - a(1)COEX(1,M) - b(1)COEX(1,L) \qquad (17) \\ &= \log S(1) \end{aligned}$$

but in this case there are none because

$$COEX(1,M) = COEX(1,L) = 0 \quad .$$

The candidate with the lowest free concentration is chosen <*SCAN1*>. Assuming $X(L)<X(M)$ we can write

$$X(L) = S(1)^{-1/b(1)}\ X(M)^{-a(1)/b(1)}\ X(H)^{-c(1)/b(1)}$$

or

$$-\log X(L) = (1/b(1))\log S(1) - (a(1)/b(1))[-\log X(M)] - (c(1)/b(1))[-\log X(H)]$$

so the elements of row L are expected to change to:

$$\mathrm{COEX}(L,M) = -a(1)/b(1)$$

$$\mathrm{COEX}(L,H) = -c(1)/b(1)$$

$$\mathrm{COEX}(L,L) = 0$$

The constant is stored in COEX(1,L) since H will never be a dependent variable and row 1 is not needed. The program finds the elements for each row of the COEX matrix with the equation

$$\mathrm{COEX}(i,j) = \mathrm{COEX}(i,j) - D(j)\mathrm{COEX}(i,L)/D(L) \qquad \text{<*SCAN1*>} \tag{18}$$

so the Lth row is generated by

$$\mathrm{COEX}(L,j) = \mathrm{COEX}(L,j) - D(j)\mathrm{COEX}(L,L)/D(L) \tag{19}$$

and for every other row i

$$\mathrm{COEX}(i,L) = 0$$

so no other row is affected. Thus the COEX matrix becomes:

$$\begin{array}{c|ccccccc|}
 & H & M & N & O & F & G & L \\
H & 1 & 0 & 0 & 0 & 0 & 0 & \log S(1)/b(1) \\
M & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
N & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
O & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
F & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
G & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
L & -c(1)/b(1) & -a(1)/b(1) & 0 & 0 & 0 & 0 & 0
\end{array}$$

Now suppose there is a second solid, $N_{a(2)}C_{b(2)}H_{c(2)}$, with formation constant S(2). N, G, and H all are independent variables, so analogous to the results of Eq.(16)

$$D(N) = a(2), \quad D(G) = b(2), \quad D(H) = c(2), \quad D(j) = 0 \quad j \neq N,G,H$$

and if $X(N) < X(G)$

$$-\log X(N) = (1/a(2))\log S(2) - (b(2)/a(2))[-\log X(G)] - (c(2)/a(2))[-\log X(H)]$$

and the COEX matrix becomes:

$$\begin{array}{c|ccccccc|}
 & H & M & N & O & F & G & L \\
H & 1 & 0 & \log S(2)/a(2) & 0 & 0 & 0 & \log S(1)/b(1) \\
M & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
N & -c(2)/a(2) & 0 & 0 & 0 & 0 & -b(2)/a(2) & 0 \\
O & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
F & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
G & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
L & -c(1)/b(1) & -a(1)/b(1) & 0 & 0 & 0 & 0 & 0
\end{array}$$

In this instance the presence of one solid has not affected the other.

For a third solid, $M_{a(3)}F_{b(3)}H_{c(3)}$, all components again are independent variables and Eq.(16) yields

$$D(M) = a(3), \quad D(F) = b(3), \quad D(H) = c(3) ,$$

and either M or F could be chosen as the dependent variable. Assume $X(M)<X(F)$. The row which expresses the dependence of X(M) upon X(F) and X(H) will be just as in the previous two solids, using an equation analagous to Eq.(18). Recall however that X(L) depends upon X(M). This dependence must be transferred to independent variables. An equation analogous to Eq.(18) is written

$$COEX(i,j) = COEX(i,j) - D(j)COEX(i,M)/D(M) \qquad (20)$$

and one finds that other rows, in addition to M, are affected because

$$COEX(i,M) \neq 0$$

for any i except M. Since

$$COEX(L,M) = -a(1)/b(1) \quad ,$$

row L and its constant will be modified

$$COEX(L,H) = -c(1)/b(1) - (c(3)/a(3))(-a(1)/b(1))$$

$$COEX(L,M) = 0$$

and

$$COEX(L,F) = 0 - (b(3)/a(3))(-a(1)/b(1)) \quad .$$

The constant for the dependence of L upon principal variables is corrected according to:

$$\begin{aligned} COEX(1,L) &= COEX(1,L) + COEX(L,M)\log K/D(M) \qquad (21) \\ &= (1/b(1))\log S(1) - (a(1)/b(1))(1/a(3))\log S(3) \end{aligned}$$

The matrix at this point takes the form shown on the following page.

The most complicated situation arises when one or more of the components in the solid is already a secondary variable. The new dependent variable then can be chosen from the independent variable(s) in the solid or from any other independent variable upon which the dependent variable(s) in the solid depend. The equations previously introduced will generate this list of candidates and when the smallest one is found they will adjust the COEX matrix.

Consider a fourth solid, $O_{a(4)}L_{b(4)}H_{c(4)}$. The possible candidates for secondary variables are those with D not equal to zero, found in a manner analagous to Eq.(16)

$$D(j) = a(4)COEX(0,j) + b(4)COEX(L,j)$$

$$D(H) = D(H) + c(4) \quad .$$

This generates

$$D(0) = a(4)$$

as usual, but

$$\begin{array}{c|ccccccc|}
 & H & M & N & O & F & G & L \\
H & 1 & \log S(3)/a(3) & \log S(2)/a(2) & 0 & 0 & 0 & \log S(1)/b(1) - a(1)\log S(1)/b(1)a(3) \\
M & -c(3)/a(3) & 0 & 0 & 0 & -b(3)/a(3) & 0 & 0 \\
N & -c(2)/a(2) & 0 & 0 & 0 & 0 & -b(2)/a(2) & 0 \\
O & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
F & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
G & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
L & -c(1)/b(1) + c(3)a(1)/a(3)b(1) & 0 & 0 & 0 & b(3)a(1)/a(3)b(1) & 0 & 0
\end{array}$$

$$D(L) = 0$$

$$D(H) = b(4)[-c(1)/b(1) + c(3)a(1)/a(3)b(1)] + c(4)$$

$$D(F) = b(4)b(3)a(1)/a(3)b(1)$$

and logK for the reaction is {as in Eq.(17)}

$$\log K = \log S(4) - a(4)\mathrm{COEX}(1,O) - b(4)\mathrm{COEX}(1,L)$$

$$= \log S(4) + [b(4)a(1)/b(1)a(3)]\log S(3) - [b(4)/b(1)]\log S(1) \quad .$$

Suppose $X(F)<X(0)$. All non-zero elements of column F are found $\{\mathrm{COEX}(i,F)\neq 0\}$, each row i is corrected, and the constant for i is updated as in Eqs.(20) and (21). For row M,

$$\mathrm{COEX}(M,F) = -b(2)/a(3)$$

so

$$\mathrm{COEX}(M,j) = \mathrm{COEX}(M,j) - D(j)\mathrm{COEX}(M,F)/D(F)$$

$$\mathrm{COEX}(1,M) = \mathrm{COEX}(1,M) + \log K\ \mathrm{COEX}(M,F)/D(F)$$

For row F,

$$\mathrm{COEX}(F,F) = 1$$

so

$$\mathrm{COEX}(F,j) = \mathrm{COEX}(F,j) - D(j)\mathrm{COEX}(F,F)/D(F)$$

$$\mathrm{COEX}(1,F) = \mathrm{COEX}(1,F) + \log K\ \mathrm{COEX}(F,F)/D(F)$$

and finally, for row L

$$\mathrm{COEX}(L,F) = b(3)a(1)/a(3)b(1)$$

so

$$\mathrm{COEX}(L,j) = \mathrm{COEX}(L,j) - D(j)\mathrm{COEX}(L,F)/D(F)$$

$$\mathrm{COEX}(1,L) = \mathrm{COEX}(1,L) + \log K\ \mathrm{COEX}(L,F)/D(F) \quad .$$

When each of these rows is corrected and row one is updated, we have the COEX matrix which is given on the following page.

$$
\begin{array}{c|ccccccc|}
 & H & M & N & O & F & G & L \\
\hline
H & 1 & \begin{array}{c}\log S(1) - \\ b(1)\log S(4)/a(1)b(4)\end{array} & \log S(2)/a(2) & 0 & \begin{array}{c}\log S(3)/b(3) - \\ a(3)\log S(1)/b(3)a(1) + \\ a(3)b(1)\log S(4)/b(3)a(1)b(4)\end{array} & 0 & \log S(4)/b(4) \\
M & \begin{array}{c}-c(1)/a(1) + \\ b(1)c(4)/a(1)b(4)\end{array} & 0 & 0 & a(4)b(1)/b(4)a(1) & 0 & 0 & 0 \\
N & -c(2)/a(2) & 0 & 0 & 0 & 0 & -b(2)/a(2) & 0 \\
O & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
F & \begin{array}{c}-c(3)/b(3) + \\ c(1)a(3)/a(1)b(3) - \\ c(4)a(3)b(1)/b(4)b(3)a(1)\end{array} & 0 & 0 & \begin{array}{c}[-a(4)a(3)b(1)/ \\ b(4)b(3)a(1)]\end{array} & 0 & 0 & 0 \\
G & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
L & -c(4)/b(4) & 0 & 0 & -a(4)/b(4) & 0 & 0 & 0
\end{array}
$$

It is of course very unlikely that four solids would precipitate among seven components, but an uncomplicated example would serve no purpose. To verify that the COEX matrix stores the correct information, each row can be generated:

$$\begin{aligned}
-\log X(M) &= -c(1)[-\log X(H)]/a(1) + b(1)c(4)[-\log X(H)]/a(1)b(4)\\
&\quad + a(4)b(1)[-\log X(0)]/b(4)a(1) + [\log S(1)]/a(1)\\
&\quad - b(1)[\log S(4)]/b(4)a(1)\\
&= -c(1)[-\log X(H)]/a(1) + b(1)\{c(4)[-\log X(H)]\\
&\quad + a(4)[-\log X(0)] - [\log S(4)]\}/a(1)b(4) + [\log S(1)/a(1)\\
-\log X(N) &= -c(2)[-\log X(H)]/a(2) - b(2)[-\log X(G)] + [\log S(2)]/a(2)
\end{aligned}$$

$$\Longrightarrow \quad X(N)^{a(2)} = X(H)^{-c(2)}\, X(G)^{-b(2)}\, S(2)^{-1}$$

$$\Longrightarrow \quad S(2) = X(N)^{-a(2)}\, X(G)^{-b(2)}\, X(H)^{-c(2)}$$

$$\begin{aligned}
-\log X(F) &= c(1)a(3)[-\log X(H)]/a(1)b(3) - c(3)[-\log X(H)]/b(3)\\
&\quad - c(4)a(3)b(1)[-\log(X(H)]/b(4)b(3)a(1)\\
&\quad - a(4)a(3)b(1)[-\log X(0)]/b(4)b(3)a(1)\\
&\quad + a(3)a(1)[\log S(4)]/b(3)b(1)b(4) - a(3)[\log S(1)]/b(3)a(1)\\
&\quad + [\log S(3)]/b(3)\\
&= -c(3)[-\log X(H)]/b(3) + a(3)\Bigl\{c(1)[-\log X(H)]/a(1)\\
&\quad - b(1)\{c(4)[-\log X(H)]/b(4) + a(4)[-\log(\underset{\uparrow}{0})]/b(4)\\
&\quad - [\log S(4)]/b(4)\}/a(1) - [\log S(1)]/a(1)\Bigr\}/b(3)\\
&\quad - [\log S(3)]/b(3)\\
-\log X(L) &= -c(4)[-\log X(H)]/b(4) - a(4)[-\log X(0)]/b(4) + [\log S(4)]/b(4)
\end{aligned}$$

$$\Longrightarrow \quad X(L)^{b(4)} = X(H)^{-c(4)}\, X(0)^{-a(4)}\, S(4)^{-1}$$

$$\Longrightarrow \quad S(4) = X(0)^{-a(4)}\, X(L)^{-b(4)}\, X(H)^{-c(4)}$$

Substituting the equation for -logX(L) into -logX(M) and -logX(F) gives

$$-\log X(M) = -c(1)[-\log X(H)]/a(1) - b(1)[-\log X(L)]/a(1) + [\log S(1)]/a(1)$$

$$\Longrightarrow \quad X(M)^{a(1)} = X(H)^{-c(1)} \; X(L)^{-b(1)} \; S(1)^{-1}$$

$$\Longrightarrow \quad S(1) = X(M)^{-a(1)} \; X(L)^{-b(1)} \; X(H)^{-c(1)}$$

and

$$-\log X(F) = -c(3)[-\log X(H)]/b(3) + a(3)\{c(1)[-\log X(H)]/a(1) + b(1)[-\log X(L)]/a(1) - [\log S(1)/a(1)\} - [\log S(3)]/b(3)$$

When -logX(M) is substituted into -logX(F)

$$-\log X(F) = -c(3)[-\log X(H)]/b(3) - a(3)[-\log X(M)]/b(3) + [\log S(3)]/b(3)$$

$$\Longrightarrow \quad X(F)^{b(3)} = X(H)^{-c(3)} \; X(M)^{-a(3)} \; S(3)^{-1}$$

$$\Longrightarrow \quad S(3) \; . = X(M)^{-a(3)} \; X(F)^{-b(3)} \; X(H)^{-c(3)}$$

Thus the equations for all four solids are regenerated.

B. MODIFICATION OF THE ERROR VECTOR AND THE Z MATRIX TO ACCOUNT FOR AMOUNTS IN SOLIDS

The calculation of concentrations of complexes, C(k) {Eq.(1)}, is done just as when no solids are present. The errors in conservation of mass for component i also are calculated as in Eq.(3), but if i is in a solid, T(i) from Eq.(2) is incomplete. An additional term is required

$$T(i) = X(i) + \sum_{k=1}^{NC} v(i,k)C(k) + \sum_{k=1}^{NSOL} u(i,k)W(k) \qquad (22)$$

where W(k) is the concentration of solid k. The concentrations of solids are not known, however. It is assumed that the error in any secondary variable, j, is due to its presence in solids, i.e.

$$-(j \text{ in solid}) = X(j) + \sum_{k=1}^{NC} v(j,k)C(k) - M(j) = Y(j) \; . \quad \text{<*SUBY*>} \qquad (23)$$

The negative amount of i in solids per j in solids was stored in COEX(j,i);

so, for independent variable i, the total amount of i becomes

$$T(i) = X(i) + \sum_{k=1}^{NC} v(i,k)C(k) + \sum_{j=1, j\neq i}^{M} COEX(j,i)Y(j) \qquad \text{<*MODIX*>} \qquad (24)$$

and this can be substituted into Eq.(3).

The variation of T(j) with respect to X(i) becomes more difficult to calculate because of the modification of T(j) {Eq.(24)} and the dependence of variations for dependent variables upon independent variables:

$$[\partial/\partial X(i)] = [\partial/\partial X(i)] + \sum_{j=1, j\neq i}^{M} [\partial X(j)/\partial X(i)][\partial/\partial X(j)] \qquad (25)$$

All dependent variables j which are functions of i will contribute to the second term. The antilog form of Eq.(15) is:

$$X(j) = COEX(1,j)\prod_{i=1}^{M} X(i)^{COEX(j,i)} \qquad (26)$$

and, easily,

$$[\partial X(j)/\partial X(i)] = X(j)COEX(j,i)/X(i) \quad . \qquad (27)$$

The presence of solids necessiates corrections to Y(i) {where i is independent} and Z(i,j) and Z(j,i), which are made in a separate subroutine <*MODIX*>. Each dependent variable j is identified by COEX(j,j) = 0. Then all independent variables i upon which j depends are found by COEX(j,i) $\neq$ 0. The term COEX(j,i)Y(j) is added to Y(i) {Eq.(24)}. The corrections to Z(k,i) {k dependent or independent} for variations due to dependent variable j are made according to

$$Z(k,i) = Z(k,i) + COEX(j,i)X(j)Z(k,j)/X(i) \qquad (28)$$

where Eqs.(25) and (27) have been used. The corrections to Z(i,k) {k dependent or independent} for independent variables i and dependent variables j are

$$Z(i,k) = Z(i,k) + COEX(j,i)Z(j,k) \qquad (29)$$

{the second term is the derivative with respect to X(k) of the last term in Eq.(24)}. When the dependence of j upon every i has been included in Y(i) and Z(k,i) and Z(i,k), Z(j,j) is set to 1, and Z(i,j) and Z(j,i) are set to zero for all i, $i\neq j$. Y(j) has not been changed and remains as the negative of the amount of j in solids. Another dependent variable is found {if any} and the procedure repeated until all dependent variables have Z(j,j) = 1.

Only errors in principal variables i are compared to total concentrations to determine whether convergence with respect to mass has been attained. The total concentration to which Y(i) is compared is the largest value of M(i) or any M(j) if X(j) is dependent on X(i) {COEX(j,i)$\neq$0} <*TESTS*>. The total concentration to which the error for H+ is compared is the largest of all M(j). The total concentration to which the error for electrons is compared is the sum of the absolute concentrations of electrons in solids and complexes <*MODIX*> {The total concentration of H+ or e- could be zero}. If convergence has not been reached, corrections to the free concentrations of independent variables must be calculated and made. This is done with the Y and Z arrays just as when there are no dependent variables <*GAUSSL*>. The equations are somewhat simplified because there are fewer independent variables. When the corrections to the independent variables have been made <*ITERTN*>, the free concentrations of dependent variables are calculated according to Eq.(26) <*MLINS1*>. If the free concentration found is larger than the total concentration, the free concentration is set equal to the total. Every five iterations the program checks that the free concentration of each dependent variable is still smaller than the free concentrations of the independent variables upon which it depends. If not, the entire COEX matrix is recalculated <*SECVAR*,*SCAN1*>. The cycle of calculating complex concentrations, C(k), error vectors, Y, and the matrix Z begins again.

C. CALCULATION OF CONCENTRATIONS OF SOLIDS AND DISSOLUTION OF UNSATURATED SOLIDS

When convergence with respect to mass has been attained, the program checks to ensure that no "precipitated" solid is undersaturated <*DISSOL*>. Undersaturation has occurred if the concentration of a solid is negative. The negative amount of each dependent variable contained in solids, Y(j), has been obtained in Eq.(23) <*SUBY*>. The stoichiometry of dependent variable j in precipitated solid k is stored as B(j,k) <*DISSOL*>. The (unknown) negative concentration of precipitated solid k is W(k).

Thus, once again there is a system of equations,

$$Y(j) = \sum_{k} B(j,k)W(k)$$

analagous to Eq.(6a). The method of solution previously described is used to solve the equations <*GAUSSL*>. Since W(k) is a negative concentration, a solid k with positive W(k) is undersaturated, and the solid k with the most positive W(k) is the most undersaturated and will be dissolved. To dissolve a solid, its removal from the list of precipitated solids is followed by a new selection of secondary variables for the remaining precipitated solids <*SOLID2*,*MIXOL2*,*SCAN1*>. Then iterations with respect to mass are repeated.

D. CHECKS FOR SUPERSATURATION OF NON-PRECIPITATED SOLIDS

If no solid dissolves, the next step is to check that the solution is not supersaturated with respect to any unprecipitated solid <*CHKSOL*>. This is done by calculating the log of the saturation ratio, S,

$$S = \sum_{i=1}^{M} v(i,k)[-\log X(i)] - \log S(k) \quad .$$

If S is greater than -0.004, the solid k is deemed to be supersaturated. The supersaturated solid {if any} with the largest S is precipitated. This is done by selecting one more secondary variable by the same procedure described in III-A and modifying the COEX matrix <*SCAN1*>. The exception to this is when a solid of the same components already exists: if the only components are a metal and hydroxide or a ligand and hydrogen and pE is not variable, the old solid is dissolved, the new one precipitated, and the entire COEX matrix is recalculated <*SOLID2*,*MIXOL2*,*SCAN1*>. This reduces the probability of creating a violation of the Gibbs phase rule. No two solids of the same two components can coexist in equilibrium with the solution, but the program is written to only screen for those containing hydrogen and a ligand or hydroxide and a metal. Precipitation of others will cause an error. This will occur only when the user has initially specified the wrong solid as precipitated {i.e. imposed the more soluble of the pair}.

If no supersaturation is found, the final two tests described in Section II for ionic strength <*IONICS*> and electroneutrality <*BALANC*> are performed and the calculations are complete. Amounts of all components in solids are included in Q(total) but the charge on positive ions will exactly equal the charge on negative ions in solids. Ligand surfaces are assumed to be uncharged in the form S-O-H. If they are charged due to complexation, this charge is part of the total solution charge calculated by Eq.(12). The errors which are reported are calculated by adding amounts of variables in solids to the errors calculated by Eqs.(2) and (3) <*SUBMOL*>. Final total concentrations are calculated by adding amounts of variables in solids to the T(i) calculated in Eq.(2).

SECTION IV. ION-ION REDOX ALLOWED

A. DEFINITION OF SECONDARY VARIABLES

The treatment of ion-ion redoxes is very similar to the treatment of solids because one dependent variable is added for each ion-ion redox reaction, m. When pE is variable, e- may be a dependent variable. An ion-ion redox equation may be expressed in a form analagous to Eq.(14),

$$\prod_{i=1}^{M} X(i)^{n(i,m)} = R(m)$$

where n(i,m) is positive for products and negative if i is a reactant, and R(m) is the reaction constant. Ion-ion redoxes will have three or four non-zero values of n(i,m). One of the variables depends upon the others or upon independent variables upon which the others depend just as with solids. The COEX matrix is determined exactly as for solids <*REDOX1*,*SCAN1*>. Dependent variables for redox equations are selected before dependent variables for solids {and mixed solids}.

After calculation of the COEX matrix is complete, procedures for computation of convergence with respect to mass are performed as described for solids, with a few minor exceptions. The value to which the error for e- is compared is the sum of the absolute number of electrons in each solid and the absolute number of electrons transferred in each reaction. Computations for cases in which a large number of electrons are transferred will thus be less precise than if few electrons are transferred. For this reason, the initial amounts of components in each redox state should be as close to the final results as possible.

Another difference for ion-ion redoxes is that the free concentration of a secondary variable in an ion-ion redox may exceed its initial total concentration. For the reaction

$$p\ X(i) \longrightarrow q\ X(j) + n\ e- \qquad (30)$$

the amount which X(i) cannot exceed is

$$M(i) + (p/q)\ M(j) \quad . \qquad <*OPTSO*>$$

That would be the total amount of i if all j were converted to i.

Ion-ion redoxes cause no problems until after convergence for mass has been achieved. If any of one component is transformed to another, the Ys for both components always will be large. However, these Ys are combined when the

Ys for dependent variables are combined with the Ys for independent variables <*MODIX*>. For solids this was adding the amount in solids; for ion-ion redoxes both the amounts transferred and in solids are added.

B. REDUCING NUMBER OF DEPENDENT VARIABLES TO EQUAL THE NUMBER OF SOLIDS

Problems arise in calculating solid concentrations <*DISSOL*>. The number of dependent variables is greater than the number of solids. The number of dependent variables must be reduced before the calculation described in III-C. In order to do this, all variables are flagged, dependent with 1 and independent with 0. One dependent variable is picked from the components of each ion-ion redox and unflagged {flag set to 0}. First the product ion is considered, then the reactant ion, and if neither is dependent, then e- is dependent and is chosen. The Ys for product i and reactant j as shown in Eq.(3) are combined according to

$$Y(i) = Y(i) + (p/q)\,Y(j)$$

$$Y(NEL) = Y(NEL) - (n/q)\,Y(j)$$

The flag for the reactant is changed to -i. For every variable flagged with 1, a row is allotted. The number of variables flagged now equals the number in solids. If the reactant is flagged and thus there is an eqution for it in the series of equations described in Section III-C, the product {negative flag} is assigned to the same row.

If a product is in a solid, the stoichiometry of the product must be multiplied by (p/q) before storage and the row for electrons must include -(n/q) as the stoichiometry of electrons in the solid. The rationale for this follows: Suppose that

$$A \longrightarrow 2\,B + 3\,e^- \qquad \{\text{with } q=2,\ p=1,\ \text{and } n=3\}$$

originally. Now, if B forms a solid,

$$B + L \longrightarrow \text{Solid}$$

when B is eliminated from the reaction,

$$(1/2)\,A + L - (3/2)\,e^- \longrightarrow \text{Solid}$$

and, indeed, the solid is viewed as

$$(p/q)\,A + L - (n/q)\,e^- \longrightarrow \text{Solid} \quad .$$

If hydrogen or hydroxide is included in either reaction their stoichiometries would be included also in describing the solid in terms of reactant. But, only stoichiometries of dependent variables are stored, so H+ stoichiometry need not be altered for the computation of solid concentrations from the amounts of dependent variables in solids.

Ion-ion redoxes have no effect upon precipitating solids or dissolving them except that their presence leads to more secondary variables.

The total concentrations, T(i), calculated for both dependent and independent i after convergence with respect to mass has been attained, are as defined in section III-D <*SUBMOL*>. They will differ from M(i) for two reasons: imprecision in the convergence test, and ion-ion redox. The latter occurs because one component was actually transformed to another and its total concentration is no longer the same. The new total concentration is used in the calculation of charge balance, Eq.(12). This equation does include both electrons {in solids and complexes} and total hydrogen {all hydrogen and hydroxide}.

C. CALCULATION OF TOTAL HYDROGEN AND ELECTRONS

The total amounts of hydrogen and electrons in solution are not conservative when ion-ion redoxes are present. Ion-ion redoxes produce and/or consume electrons and possibly hydrogen. For reporting, these changes are readily calculated from the changes in total concentration of each product in an ion-ion redox reaction multiplied by the H+ or e- stoichiometry and divided by its own <*SUBMOL*>.

LIST OF VARIABLES FOR REDEQL-UMD

PARAMETER STATEMENT

NAME	STANDARD VALUE	DESCRIPTION	POSSIBLE VALUES
MXM	20	Maximum number of metals (including H+)	1 - 50
MXL	30	Maximum number of ligands (including OH-)	1 - 99
MXT	50	MXM + MXL	2 - 149
MXN	10	Maximum number of cases	1 - 10
MXC	750	Maximum number of complexes	1 - ...
MXS	100	Maximum number of solids	1 - 999
MXR	20	Maximum number of redox reactions	1 - 60
MMC	100	Maximum number of mixed complexes (must be less than MXC)	1 - ...
MXX	10	Maximum number of mixed solids.	1 - 60
MMM	2500	MXT * MXT	
MNN	500	MXT * MXN	
MXG	60	MXX * 6	
MML	600	MXM * MXL	

The parameter statement is used in the MNF compiler to set the size of the arrays. This flexibility is desirable for increasing or decreasing the capacity of the program. Smaller capacity means a program that requires less core and runs more rapidly. Larger capacity would allow the solution of more complex equilibrium problems.

For computers which do not support the parameter statement, it must be removed from each subroutine and the parameters listed above must be replaced with definite values.

VARIABLE LIST

The following is a list of the global variables used in REDEQL-UMD. Abbreviations used as columnar headings are: D, defined or set; M, modified; U, used; R, reported.

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
ACOF	R	IONS	ACOF = ACON*(0.08I-SQRTI/(1+1.51SQRTI)) for Magnuson activity coefficients, log gamma = ACOF + ACOF2. ACOF = ACON*(0.3I-SQRTI/(1+SQRTI)) for Davies activity coefficients, log gamma = ACOF + ACOF2.	IONCOR		ACTLOG	TOTCC		
ACOF2	R	IONS	ACOF2 = 0.09I for Magnuson activity coefficients. = 0.0 for Davies activity coefficients.	IONCOR		ACTLOG	TOTCC		
ACON	R	IONS	Temperature dependent coefficient in activity coefficient calculation (ACOF, ACT1).	INMAT (T-1)		IONCOR			
ACT1	R	IONS	ACT1 = ACON(0.08(I-IO)-SQRTI/(1+1.51SQRTI) + SQRTIO/(1+1.51SQRTIO)) for Magnuson coefficents, ACT1 = ACON(0.3(I-IO)-SQRTI/(1+SQRTI)+SQRTIO/(1+SQRTIO) for Davies activity coefficients, where IO=old ionic strength and delta log gamma = ACT1 + ACT2.	IONCOR		GAMA			
ACT2	R	IONS	ACT2 = 0.09(I-IO) for Magnuson equation. ACT2 = 0 for Davies equation.	IONCOR		GAM GAMA			
ALLCO2	R	NUMB	Flag for CO2 considered in some case. 0 if in no case. Greater than zero if in any case.	INMAT		OUT1			GE 0
ALLN2	R	WRED	Flag for N2 oxidizing to NO3- considered in any case. 0 if in no case. Greater than 0 if in any case.	INMAT		INRED OUT1	OUTCST OUT1N		GE 0
ALPH(5)	R	ADSO	ALPH(i) = adjusting parameter alpha(i) for strength of Nernstian potential for surface i.	INADS (U-13)		ADSORP	OUTSOL		0 - 1
CHGM(MXT)	I	CONC	CHGM(i) = charge on the ith variable.	INCONC INMAT		ADSORP ALLOTC BALANC INRED	IONCOR IONICS OUT1 TOTCC		-5 - 5
CHOH(MXC)	I	COMP	CHOH(K) = stoichiometric coefficient of H+ (OH- if negative) in the kth complex.	ALLOTC INRED (T-7)		ADSORP CO3ALK IONCOR	SUBXC SUBY SUBZ	OUTCST OUTMC OUT17	
CHR(MXC)	I	CONC	CHR(K) = charge on the kth complex	ALLOTC INRED		IONCOR IONICS	OUT17 OUTMC		-5 - 5

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
CK(MXC)	R	COMP	CK(k) = log K where K is the formation constant for the kth complex.	ALLOTC INRED (T-7)	ADSORP IONCOR	SUBXC		OUTCST OUTMC	
CL(MXC)	I	COMP	CL(k) = stoichiometric coefficient of ligand in the kth complex.	ALLOTC INRED (T-7)		CO3ALK IONCOR SUBXC	SUBY SUBZ	OUT17 OUTCST OUTMC	
CL2(MMC)	I	COMP	CL2(J) = stoichiometric coefficient of third ion in jth mixed complex.	ALLOTC		CO3ALK IONCOR SUBXC	SUBY SUBZ	OUTMC	
CLIG(MXC)	I	COMP	CLIG(K) = variable number of ligand in kth complex.	ALLOTC INRED		ADSORP CO3ALK IONCOR OUT17 OUTCST	OUTMC SUBXC SUBY SUBZ		1 - NOH
CLIG2(MMC)	I	COMP	CLIG2(J) = variable number of third ion in jth mixed complex.	ALLOTC		CO3ALK IONCOR OUTMC SUBXC	SUBY SUBZ		1 - NOH
CM(MXC)	I	COMP	CM(K) = stoichiometric coefficient of metal in kth complex.	ALLOTC INRED (T-7)		ADSORP CO3ALK IONCOR	SUBXC SUBY SUBZ	OUT17 OUTCST OUTMC	
CMET(MXC)	I	COMP	CMET(K) = variable number of metal in kth complex.	ALLOTC INRED		ADSORP CO3ALK IONCOR OUT17 OUTCST	OUTMC SUBXC SUBY SUBZ		1 - NOH
COEX(MXT,MXT)	R	BIGD	COEX(i,j) = a in the following equation for the free concentration of secondary variable i in terms of other free concentrations. X(i) = K(X(j)**a)(X(k)**b) COEX(1,i) = negative log of the constant K in the above equation.	SCAN1 SECVAR		DISSOL ITERTN MIXOL2 MLINS1 MODIX	REDOX1 SOLID2 SUBJAC TESTS		
COML(MXM,MXL)	R	BIGD	COML(i,j) = concentration of ligand j in complexes of ith metal with jth ligand. COML(1,1) = COEX(1,13)	OUTCST		OUT139			

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
COMM(MXM,MXL)	R	BIGD	COMM(i,j) = concentration of metal i in complexes of ith metal with jth ligand. COMM(1,1) = COEX(1,1)	OUTCST		OUT139			
CONST(12)	R	CNST	CONST(i) = log K for ith solid or complex of a data line.	INADS (U-13,14) INMAT (T-5) INMIXC (T-11)		ALLOTC ALLOTS			
DMIN	R	MPRM	0.5%, the minimum percentage of a variable that will be reported.	BLOCKDATA		OUT139 OUT140	OUT141		
EMAX(MXT)	R	XTRA	EMAX(i) = maximum possible total concentration of variable i. EMAX = TOTM(i) if no ion-ion redoxes include this variable.	TOTCC	OPTSO	MLINS1			
ENIT(MXN)	R	XTRA	ENIT(i) = initial concentration (moles/l) of electrons in solids and complexes for case i.	INMAT (U-8)		OPTSO	TOTCC	OUTIN	
EPS	R	MPRM	0.0001. Convergence with respect to mass has been attained when ABS(Y(i))/TESTY(i) LT EPS for every independent variable i.	BLOCKDATA		CVTEST			
EXH	R	CONC	Free concentration of H+, EXM(1).	TOTCC	IONICS ITERTN	SUBY	SUBZ	OUT1	
EXM(MXT)	R	CONC	EXM(i) = free concentration of variable i, (moles/l).	INCONC OPTSO TOTCC	ITERTN MLINS1	CO3ALK IONICS MODIX OUT139	SCAN1 SUBJAC SUBY SUBZ	OUT1	
EXOH	R	CONC	Free concentration of OH-, 10**(XH-WK).	TOTCC	IONICS ITERTN	SUBY	SUBZ		
FAC	R	MPRM	Factor for converting concentration to a percent of total concentration: 100/TOTM(i).	OUT139		OUT140	OUT141		
GAMTBL(5)	R	IONS	GAMTBL(i) = change in log of activity coefficient for an ion of charge i for a new ionic strength.	IONCOR		GAM	IONICS		

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
I17	I	OUTC	0: complexes will be reported. 1: complexes will not be reported.	INMAT (U-1)		SUB3			0,1
IACT	I	OUTC	0: complexes reported as moles/l. 1: complexes reported as activities and as moles/l.	INMAT (U-1)		OUT17			0,1
IBEG	I	CLEK	Number of first solid or complex in a group having the same metal and ligand components.	COLECT		OUT17	OUTCST OUTSOL		1 - NC or NS
ICHG	I	INIT	Charge on ion used for charge balance adjustment.	INMAT (U-1)		BALANC			-5 - 5
ICON	I	OUTC	0: thermodynamic data printed uncorrected and corrected. 1: thermodynamic data printed uncorrected only. 2: thermodynamic data not printed.	INMAT (U-1)		SUB2	SUB3		0,1,2
IEND	I	CLEK	Number of solid or complex in a group having the same metal and ligand components.	COLECT		OUT17	OUTCST OUTSOL		1 - NC or NS
IGAM	I	IONS	0: Magnuson activity coefficients. 1: Davies activity coefficients.	INMAT (U-1)		IONCOR	OUTCST		0,1
IJAC	I	OUTC	Selects interaction intensity/capacity output 0: no output. 1: capacities printed. 2: intensities printed. 3: capacities and intensities printed.	INMAT (U-1)		SUB3	SUBJAC		0,1,2,3
ION	I	NUMB	0: ionic strength supplied by the user remains fixed for all cases. 1: ionic strength is assumed 0 and is recalculated if necessary each time convergence with respect to mass is attained.	INMAT (U-1)		OUT1N SUB2	SUB3		0,1
ISOLUB(MXM,MXL)	I	SOLN	ISOLUB(i,j) = m if solid m is the only solid formed from metal i and ligand j. If m GT 1000, two precipitated solids n and p, are formed from metal i and ligand j, and m = 1000*n + p.	INMAT INRED OPTSO TOTCC	CHKSOL DISSOL	OUT139			0 - 1000* (NS+1)

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
ITER	I	MPRM	Number of iterations with respect to mass completed for current case.	TOTCC	CVTEST	DISSOL MLINS1	SUB3		0 - ITMAX
ITMAX	I	MPRM	200, maximum number of iterations allowed before execution is halted.	BLOCKDATA		CVTEST	DISSOL		
JFIX	I	INIT	Reference number of ion whose total concentration is to be adjusted for charge balance.	INMAT (U-1)		BALANC			
KLIG(MXS)	I	SOLC	KLIG(k) = variable number of ligand in kth solid.	ALLOTS INMAT INRED		CHKSOL DISSOL IONCOR OPTSO OUT1	OUTIN OUTSOL SCAN1 SOLID2 SUBMOL		1 - NOH
KLR(MXR)	I	REDX	KLR(i) = stoichiometry of second ion in redox reaction i.	INRED (U-7)		DISSOL IONCOR OPTSO	REDOX1 SUBMOL	OUTCST	
KMET(MXS)	I	SOLC	KMET(k) = variable number of metal in kth solid.	ALLOTS INMAT INRED		CHKSOL DISSOL IONCOR OPTSO OUT1	OUTIN OUTSOL SCAN1 SOLID2 SUBMOL		1 - NOH
KMIX(MXX)	I	MIXS	KMIX(i) = status of ith mixed solid 0: allowed. 1: precipitated. -1: cannot precipitate	INMIX	CHKSOL DISSOL	SECVAR	OUT139		-1,0,1
KMR(MXR)	I	REDX	KMR(i) = stoichiometry of first ion in redox reaction i.	INRED (U-7)		DISSOL IONCOR OPTSO	REDOX1 SUBMOL	OUTCST	
KPREC(MXS)	I	SOLN	KPREC(k) describes status of the kth solid -1: allowed. 0: cannot precipitate. 1: precipitated.	INMAT INRED OPTSO TOTCC	CHKSOL DISSOL	OUT1	SECVAR		-1,0,1
KREDOX	I	WRED	0: no redox allowed. 1: redox allowed at fixed potential. 2: redox allowed with varied potential.	INMAT (U-1)		CHKSOL INRED ITERTN OPTSO OUT1	OUTIN SUB3 SUBZ TOTCC		0,1,2

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
KSHIX(MXX)	I	MIXS	KSHIX(k) = stoichiometric coefficient of H+ (OH- if negative) in the kth mixed solid.	INMIX (T-9)		CHKSOL IONCOR	MIXOL2 SUBMOL	OUTSOL	
KSHOH(MXS)	I	SOLC	KSHOH(k) = stoichiometric coefficient of H+ (OH- if negative) in the kth solid	ALLOTS INMAT (T-5) INRED (T-7)		CHKSOL DISSOL IONCOR	SCAN1 SOLID2 SUBMOL	OUT1 OUTIN OUTSOL	
KSL(MXS)	I	SOLC	KSL(k) = stoichiometric coefficient of ligand in the kth solid.	ALLOTS INMAT (T-5) INRED (T-7)		CHKSOL DISSOL IONCOR OUT139	SCAN1 SOLID2 SUBMOL	OUT1 OUTIN OUTSOL	
KSM(MXS)	I	SOLC	KSM(k) = stoichiometric coefficient of metal in the kth solid.	ALLOTS INMAT (T-5) INRED (T-7)		CHKSOL DISSOL IONCOR OUT139	SCAN1 SOLID2 SUBMOL	OUT1 OUTCST OUTIN OUTSOL	
KSMIX(MXX,6)	I	MIXS	KSMIX(k,m) = stoichiometric coefficient of mth ion in kth mixed solid.	INMIX (T-9)		CHKSOL DISSOL IONCOR	MIXOL2 OUT139 SUBMOL	OUTSOL	
LI(100)	I	TINN	LI(i) = variable number of ligand with reference number i. LI(i) = 0 means ligand is not in the system.	INMAT		CHEKML CHKSOL CO3ALK INMIX	INMIXC INRED OUTIN TOTCC		0 - NOH
LIG(12)	I	CNST	LIG(i) = stoichiometry of ligand in the ith solid or complex of a data line.	INADS (U-14) INMAT (T-5) INMIXC (T-11)		ALLOTC	ALLOTS		
LIG	I	CLEK	Variable number of ligand in the group having same metal and ligand components.	COLECT		OUT17	OUTCST OUTSOL		1 - NOH
LIG2(12)	I	CNST	LIG2(i) = stoichiometry of third ion in ith mixed complex.	INMIXC (T-11)		ALLOTC			

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
LIMIT	I	CLEK	Highest value IEND can assume for type of complexes or solids being scanned.	OUT17 OUTCST OUTSOL		COLECT			1 - NC or NS
LREDOX(MXR)	I	REDX	LREDOX(k) = variable number of second ion in kth ion-ion redox.	INRED		DISSOL IONCOR OPTSO OUTCST	REDOX1 SUBMOL		1 - NOH
MCMIX	I	MIXS	Number of mixed complexes in the system.	ALLOTC INMIXC		CO3ALK IONCOR OUT17	OUTCST SUBXC SUBY		1 - MMC
MCRED	I	MIXS	Number of redox complexes in the system.	INRED		OUT17 OUTCST	SUBXC SUBY		1 - MXR
MET(12)	I	CNST	MET(i) = stoichiometry of metal in ith solid or complex of a data line.	INADS (T-14) INMAT (U-5) INMIXC (U-11)		ALLOTC ALLOTS			
MET	I	CLEK	Variable number of metal in a group having the same metal and ligand components.	COLECT		OUT17 OUTCST	OUTSOL		1 - NOH
MGW	I	INIT	Flag for concentration units for input 0: -log M. 1: mg/l.	INMAT (U-1)		INCONC OUTCNC	OUTIN		0,1
MI(50)	I	TINN	MI(i) = variable number of metal with reference number i. MI(i) = 0 means metal i is not in the system	INMAT		CHEKML INADS INMIX	INMIXC INRED		0 - NOH
MIX(MXX,6)	I	MIXS	MIX(k,n) = variable number of the nth ion in the kth mixed solid.	INMIX		CHKSOL DISSOL IONCOR MIXOL2	OUT139 OUTSOL SUBMOL		1 - NOH
MIXC(MXT)	I	CONC	MIXC(i) flags whether variable i is in any mixed complex. 0: variable i is not in any mixed complex. 1: variable i is in at least one mixed complex.	INMIXC		OUT139			0,1

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
MMLL	I	REDX	Number of ion-ion redoxes.	INRED		BALANC DISSOL IONCOR OPTSO OUTCST	REDOX1 SECVAR SUBMOL SUBY		0 - MXR
MO(MXT)	I	INIT	MO(I) = reference number of ith variable (a metal if i LE NM).	INCONC (U-3,4) INMAT		DEFALT DISSOL		OUT1 OUT17 OUTCNC OUTCST OUTSOL	1 - 100
MREDOX(MXR)	I	REDX	MREDOX(k) = variable number of first ion in kth ion-ion redox.	INRED		DISSOL IONCOR OPTSO	OUTCST REDOX1 SUBMOL		1 - NOH
MSOLID(MXT)	I	CONC	MSOLID(k) = solid number of kth precipitated solid (if k LE NSOL) or (k-NSOL)th precipitated mixed solid (if k GT NSOL).	DISSOL INMAT INMIX INRED		OUT1 OUTIN	SUBMOL		1 - (NOH - 1)
MSRED	I	REDX	Number of redox solids.	INRED		OUTSOL			1 - MXR
N2V	I	BIGD	Number of secondary variables.	SECVAR	CHKSOL DISSOL	SUB3	SUBJAC		1 - (NOH - 1)
NALK	I	NUMB	Flag for inorganic carbon: 0: total inorganic carbon given as CO3(2-). 1: total carbonate alkalinity given as CO3(2-). 2: -log partial pressure of CO2 given as well as total (large) inorganic carbon.	INMAT (U-1)		CHKSOL OUTIN	SUB3		0,1,2
NAMEM(MXX)	I	MIXS	NAMEM(k) = name of the kth mixed solid.	INMIX (T-9)				CHKSOL DISSOL OUT1 OUT139 OUTIN OUTSOL SCAN1	
NAMER(MXR)	I	REDS	NAMER(k) = name of the kth redox reaction.	INRED (T-7)				OUTIN	

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
NAMML(MXT)	I	NAME	NAMML(i) = name of the ith variable.	INCONC (T-2,3) INMAT (T-2,3) INADS (U-13)	OUTJAC			BALANC CHKSOL DISSOL OUT1 OUT17 OUT139 OUT140 OUTCNC OUTCST OUTIN OUTJAC OUTMC OUTSOL SCAN1	
NBMIX	I	MIXS	1) 0: no mixed solids considered. 1: mixed solids considered. 2: mixed solids considered with some present or disallowed. 3: mixed solids checked for saturation only.	INMAT (U-1)		INMIX			0,1,2,3
			2) Number of mixed solids precipitated.	INMIX	CHKSOL DISSOL	OUT1 OUT139 OUTIN	SECVAR SOLID2 SUBMOL		0 - NMIX
NC	I	COMP	Number of complexes: pointer to slot allotted to last complex.	INMAT	ALLOTC INRED	ADSORP CO3ALK INMIXC OUT17	OUTCST SUBXC SUBY SUBZ	OUTIN	1 - MXC
NCASE	I	CASE	Number of the case currently being treated.	UMDEQ		OPTSO OUT1	SUB3 TOTCC	PAGE	1 - MXN LE 10
NCASES	I	INIT	Number of cases treated.	INMAT (U-1)		UMDEQ DEFALT INCONC	OUTCNC OUTIN		
NCMIX	I	COMP	Points to the last mixed complex among all complexes.	INMIXC		ADSORP CO3ALK IONCOR IONICS	OUT17 OUTCST OUTMC SUBZ		1 - NC

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
NCRED	I	REDC	Points to the last redox complex among all complexes.	INRED		CO3ALK IONCOR OUTNC	SUBXC SUBY SUBZ		1 - NC
NCREG	I	REDC	Points to the last regular complex among all complexes.	INMAT		INRED OUT17 OUTCST	SUBXC SUBY SUBZ		1 - NC
NEL	I	CONC	Variable number for electrons(= NOH).	INMAT		DISSOL ITERTN MLINS1 MODIX OPTSO OUT1	REDOX1 SOLID2 SUBMOL SUBY TOTCC		NOH
NELC(MXR)	I	REDC	NELC(j) = number of electrons produced in the formation of the jth redox complex.	INRED (T-7)		OUT17 SUBXC	SUBY SUBZ	OUTCST	
NELEC(MXR)	I	REDX	NELEC(k) = number of electrons produced in the kth ion-ion redox reaction.	INRED (T-7)		DISSOL REDOX1	SUBMOL	OUTCST	
NELS(MXR)	I	SOLN	NELS(i) = number of electrons produced in the formation of the ith redox solid.	INRED (T-7)		CHKSOL DISSOL	SOLID2 SUBMOL	OUTSOL	
NES	I	BIGD	Variable number of last ligand excluding OH- and e- (= NOH - 1).	INMAT		BALANC CO3ALK OUT1 OUT139 OUTIN SUB3	SUBMOL SUBY TESTS TOTCC OUTJAC		NOH - 1
NPL	I	NUMB	Variable number of first ligand (= NM + 1).	INMAT		BALANC	OUTIN		NM + 1
NPRST	I	BIGD	Variable number of first ion of unknown free concentration.	INMAT		CVTEST ITERTN	SUB3		1,2
NHRED	I	REDX	NHRED(k) = stoichiometry of H+ (OH- if negative) ion in kth ion-ion redox reaction.	INRED (T-7)		IONCOR REDOX1	SUB3 SUBMOL	OUTCST	
NIDS	I	OUTC	Number of initially precipitated solids plus number of solids not allowed to precipitate. -1: no solids may precipitate.	INMAT		OUTIN			-1 - 26

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
NIQ	I	BIGD	Variable number of last ion excluding OH-. (= NOH - 1 or NEL)	INMAT		DISSOL INVERT MIXOL2 MLINS1 MODIX OUTJAC	REDOX1 SCAN1 SECVAR SOLID2 SUBJAC SUBZ		
NIX(MXX)	I	MIXS	NIX(k) = number of ions (excluding H+/OH-) in kth mixed solid.	INMIX		CHKSOL DISSOL IONCOR MIXOL2	OUT139 OUTSOL SUBMOL		1 - 6
NL	I	NUMB	Number of ligands and ligand surfaces including OH- (= NL3 + NLADS).	INMAT		OUT139	OUTCST		NL3 + NLADS
NL2	I	ADSO	Variable number of last ligand excluding OH- and surfaces (= NM + NL3 - 1).	INMAT		ADSORB INADS	OUTIN		NL3 - 1
NL3	I	NUMB	Number of ligands including OH-.	INMAT (U-1)		IONICS	OUTIN		1 - MXL
NLADS	I	ADSO	Number of surfaces which adsorb metals.	INMAT (U-1)		ADSORB INADS OUTIN	SUB1 SUB3		0 - 5
NLAST	I	BIGD	Variable number of last ion of unknown concentration. (= NOH - 1 or NEL)	INMAT		CVTEST DISSOL ITERTN MLINS1	MODIX SCAN1 SUB3		NOH - 1 or NEL
NM	I	NUMB	Number of metals including H+.	INMAT (U-1)		BALANC CHKSOL DISSOL INRED IONICS	OPTSO OUT139 OUTCST OUTIN SUB3 TOTCC		1 - MXM
NMIX	I	MIXS	Number of mixed solids.	INMIX		CHKSOL DISSOL IONCOR OUT139	OUTCST OUTSOL SECVAR SUBMOL		1 - MXX

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
NOH	I	CONC	Variable number of OH- (= NM + NL).	INMAT		CHKSOL DISSOL OUTIN	SUBJAC SUBZ		2 - MXT
NPH	I	NUMB	0: pH fixed for each case. 1: pH computed for each case.	INMAT (U-1)		IONICS ITERTN OUT1 OUTIN	SUB3 SUBMOL TOTCC		0,1
NPROT(12)	I	CNST	NPROT(i) = stoichiometry of H+ in ith solid or complex of data line.	INADS (U-14) INMAT (T-5) INMIXC (T-11)		ALLOTC ALLOTS			
NREDOX	I	WRED	Number of redox reactions.	INRED		OUTIN			0 - MXR
NRS	I	OUTC	0: degree of saturation will be reported. 1: degree of saturation will not be reported.	INMAT (U-1)		SUB3			0,1
NS	I	SOLN	Number of solids: pointer to slot alloted to the last solid.	INMAT	ALLOTS INRED	CHKSOL DISSOL IONCOR OUTCST	OUTIN SECVAR SUBMOL		0 - MXS
NSOL	I	SOLN	Number of precipitated solids.	INMAT	CHKSOL DISSOL INRED OPTSO TOTCC	INMIX OUT1 OUTIN	SECVAR SUBMOL		0 - (NOH - 1)
NSREG	I	SOLN	Number of regular solids: pointer to slot alloted to last regular solid.	INMAT		CHKSOL DISSOL INRED OPTSO OUT1	OUTSOL SOLID2 SUBMOL		1 - NS
OLDSTR	R	ION	Old ionic strength.	SUB2 IONICS		IONCOR			
PCO2(MXC)	R	INIT	PCO2(i) = CO2 partial pressure in case i (-log atm). 0: CO2 not considered for case i.	INMAT (U-6)		TOTCC		OUT1 OUTIN	

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
PCO2CT	R	CONC	log K for the reaction: 2H+ + CO3(2-) ---> CO2(g) + H2O.	INMAT (T-4)	IONCOR	TOTCC		OUT1	
PE	R	CONC	-log electron activity for current case, pE.	INMAT (U-7) OPTSO	ITERTN	CHKSOL	SUBXC	OUT1	-10 - 20
PEM(MXN)	R	WRED	PEM(i) = electron activity for case i, pE.	INMAT (U-7)		OPTSO		OUTIN	
PN2(MXT)	R	WRED	PN2(i) = N2 partial pressure for case i, -log atm 0: no N2 considered.	INRED (U-11)		OPTSO		OUT1 OUTIN	
PN2CST	R	WRED	Log K for the reaction: 2NO3- + 12H+ + 10e- ---> N2(g) + 6H2O.	INRED (T-7)	IONCOR	OPTSO		OUT1 OUTCST	
PZC(5)	R	ADSO	PZC(i) = pH of ZPC of the ith adsorbing surface.	INADS (U-13)		ADSORP			0 - 14
REDCST(MXR)	R	REDX	REDCST(i) = log K for the ith ion-ion redox reaction.	INRED (T-7)	IONCOR	REDOX		OUTCST	
SAT(MXS)	R	SOLC	SAT(i) = log of the ion product over the solubility product of the ith solid.	CHKSOL				OUTSAT	
SATX(MXX)	R	MIXS	SATX(j) = log of the ion product over the solubility product of the jth mixed solid.	CHKSOL				OUTSAT	
SOIMP(MXS)	R	SOLC	SOIMP(k) = concentration of kth solid (zero if kth solid not precipitated).	SUBMOL		OUT140		OUT1	
SOLHY(MXS)	R	SOLC	SOLHY(m) = log K for formation of mth solid.	ALLOTS INMAT INRED (T-7) OPTSO TOTCC	IONCOR	CHKSOL OUTSAT	SOLID2	OUTCST	
SOLIX(MXS)	R	MIXS	SOLIX(m) = log K for formation of the mth mixed solid.	INMIX (T-9)	IONCOR	CHKSOL	MIXOL2	OUTCST	

VARIABLE	TYPE	COMMON BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
SOMIX(MXX)	R	MIXS	SOMIX(k) = concentration of kth mixed solid.	SUBMOL		OUT139		OUT1	
STOTHM	R	TOTC	Total H+ less total OH- in solution: soluble TOTH	SUBMOL				OUT1	
STRNGH	R	IONS	Ionic strength	INMAT (U-2)	IONICS	BALANC	IONCOR	OUT1 OUTIN	
TESTE	R	TOTC	TESTY(NEL) = Sum of electrons in redox complexes.	SUBY		MODIX			
TESTY(MXT)	R	TOTC	TESTY(i) = test concentration of variable i. Error, ABS(Y(i)), of every principal variable i must be less than EPS*TESTY(i) for convergence with respect to mass to be achieved.	TESTS MODIX		CVTEST			
TMIN(MXT,MXN)	R	INIT	TMIN(i,j) = total concentration of variable i in case j, -log M. TMIN(1,j) = fixed pH for case j.	INCONC (U-3,4) INMAT (U-5)		TOTCC		OUTIN OUTCNC	
TMINW(MXT,MXN)	R	INIT	TMINW(i,j) = total concentration of variable i in case j, mg/l.	INCONC (U-3,4)				OUTCNC	
TOTE	R	TOTC	Total concentration of electrons (in complexes and solids).	TOTCC OPTSO	SUBMOL	SUBY	BALANC	OUT1	
TOTH	R	TOTC	Total H+ - total OH-, moles/l.	TOTCC	SUBMOL	SUBY		OUT1	
TOTM(MXT)	R	TOTC	TOTM(i) = total concentration of variable i, moles/l.	TOTCC	SUBMOL	BALANC CO3ALK OPTSO OUT139	SUBJAC SUBY TESTS	OUT1	
WK	R	CONC	-log ionization constant of water: 14.0 at zero ionic strength, 25 deg C.	INMAT	IONCOR	IONICS ITERTN	TOTCC	OUT1	
XC(MXC)	R	COMP	XC(k) = concentration of kth complex, moles/l.	SUBXC		IONICS CO3ALK OUT141	OUTCST SUBY SUBZ	OUT17 OUTMC	
XH	R	COMP	-log concentration of free H+, XM(1).	TOTCC	IONICS ITERTN	ADSORP CHKSOL	OUTSAT SUBXC	OUT1	

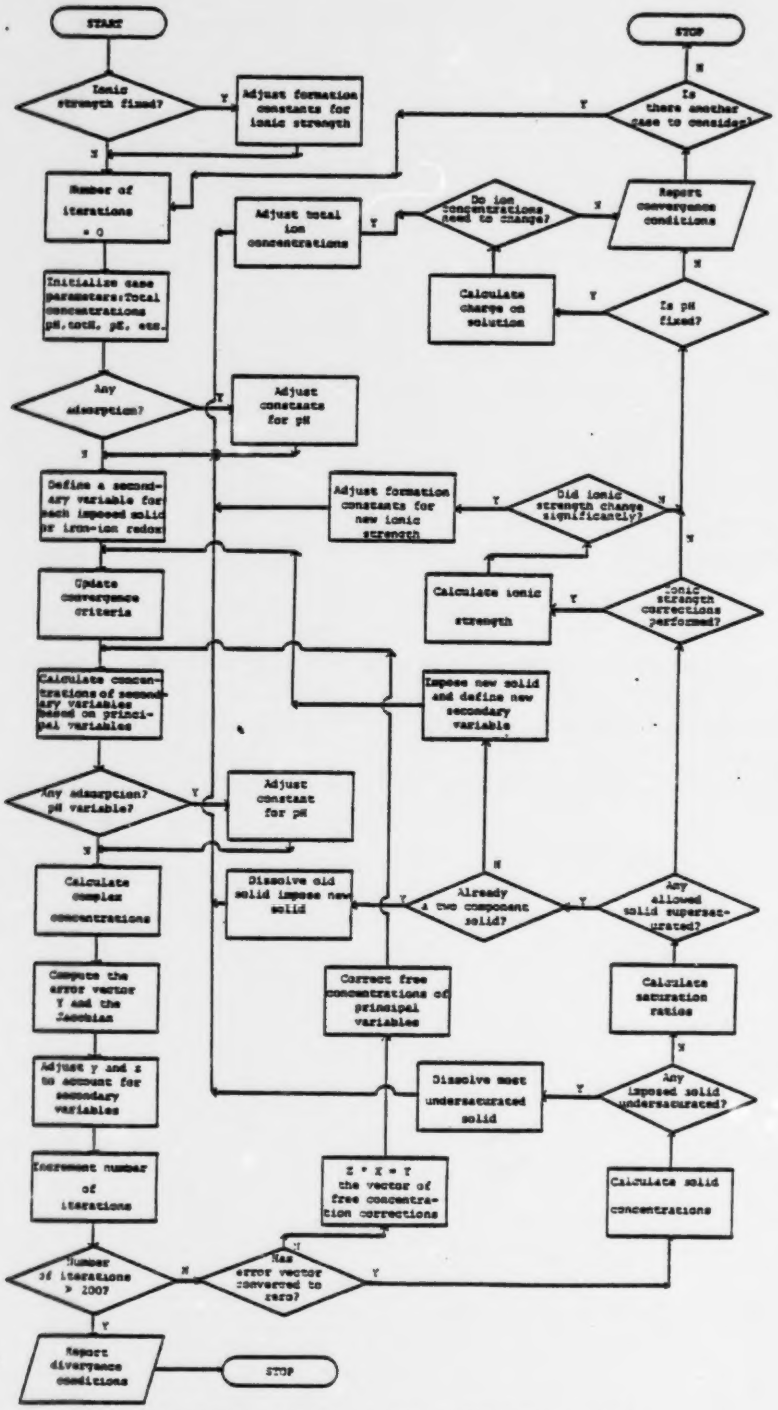
VARIABLE	COMMON TYPE BLOCK	DESCRIPTION	D	M	U	U	R	NORMAL RANGE
XM(MXT)	R CONC	XM(i) = -log concentration of free variable i.	INCONC TOTCC	ITERTN MLINS1 OPTSO	CHKSOL	SUBXC	OUT1	
Y(MXT)	R BIGD	1) error vector; preliminary vector defined in SUBY; used in SUBZ to compute preliminary Jacobian; modified to account for solids in MODIX; tested for convergence to zero in CVTEST.	CO3ALK SUBY	MODIX	CVTEST GAUSSL	SUBZ		
		2) dependence vector: non-zero if i is independent variable in the solid or ion-ion reaction or if a dependent variable in the reaction depends on i.	MIXOL2 REDOX1 SOLID2		SCAN1			
		3) in GAUSSL Y is the solution vector X and the vector of dependent variables Y in the equation Z * X = Y.						
		a) when called by SUB3, Y is Y as defined in 1 and becomes the vector of free concentration adjustments (see 1).	(see 1)	GAUSSL	ITERTN			
		b) when called by DISSOL, Y is the preliminary error vector for secondary variables and becomes the vector of negative solid concentrations.	SUBY	DISSOL GAUSSL	SUBMOL			
		4) Y is the complete error vector when computed in SUBMOL and is reported in OUT1.	SUBY	SUBMOL	BALANC		OUT1	
		5) Y is used for storage in SUBJAC.	SUBJAC					
Z(MXT,MXT)	R BIGD	1) Jacobian matrix Z(i,j) = dYi/dXj: preliminary matrix computed in SUBZ, modified to account for solids and ion-ion redoxes in MODIX and used to compute adjustments of free concentrations in GAUSSL.	SUBZ	CO3ALK MODIX	GAUSSL			
		2) Fraction of ith secondary variable in jth imposed solid. Calculated in DISSOL and used in GAUSSL to calculate concentrations of solids.	DISSOL		GAUSSL			
		3) Matrix of interaction capacities or intensities.	SUBZ	INVERT MODIX SUBJAC			OUTJAC	

FLOW DIAGRAMS FOR REDEQL-UMD

Modification of a program of this size requires both knowledge of the overall flow of computations and a detailed, subroutine-by-subroutine analysis of program logic.

On the next page is given a diagram of the computational flow of the program. No relationships to specific subroutines are intended, but an understanding of flow from this diagram may make the later use of each subroutine more easily understood.

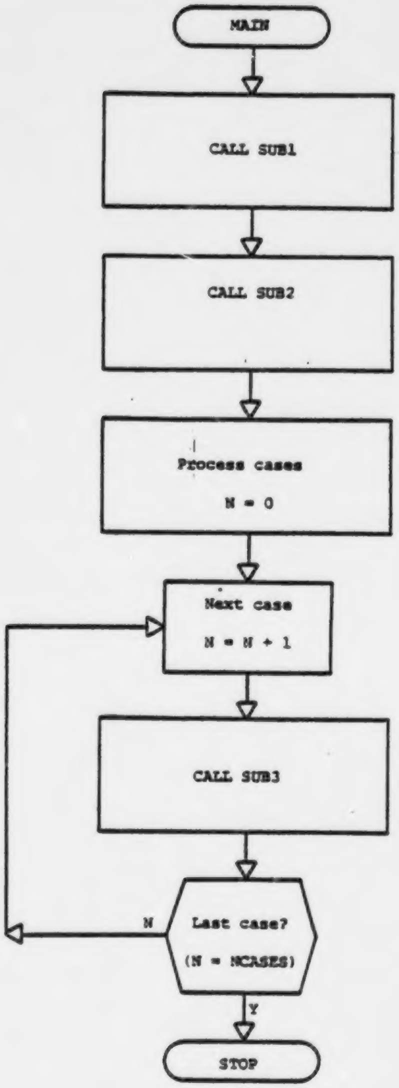
Following the computational flow diagram are flow diagrams for the main program and all subroutines in the program, each preceded by a short description of its function. First diagrammed is the MAIN program followed by the three subroutines called by MAIN (SUB1, SUB2, SUB3). All other subroutines are in alphabetical order.


START
Ionic strength fixed?
Adjust formation constants for ionic strength
Number of iterations = 0
Initialize case parameters: Total concentrations pH, totH, pE, etc.
Any adsorption?
Adjust constants for pH
Define a secondary variable for each imposed solid or iron-ion redox
Update convergence criteria
Calculate concentrations of secondary variables based on principal variables
Any adsorption? pH variable?
Adjust constant for pH
Calculate complex concentrations
Compute the error vector Y and the Jacobian
Adjust y and z to account for secondary variables
Increment number of iterations
Number of iterations > 200?
Report divergence conditions
STOP
Has error vector converged to zero?
Z * X = Y the vector of free concentration corrections
Correct free concentrations of principal variables
Dissolve old solid impose new solid
Already a two component solid?
Impose new solid and define new secondary variable
Dissolve most undersaturated solid
Any imposed solid undersaturated?
Calculate solid concentrations
Calculate saturation ratios
Any allowed solid supersaturated?
Ionic strength corrections performed?
Calculate ionic strength
Did ionic strength change significantly?
Adjust formation constants for new ionic strength
Is pH fixed?
Calculate charge on solution
Do ion concentrations need to change?
Adjust total ion concentrations
Report convergence conditions
Is there another case to consider?
STOP

Calculational flow for REDEQL-UMD

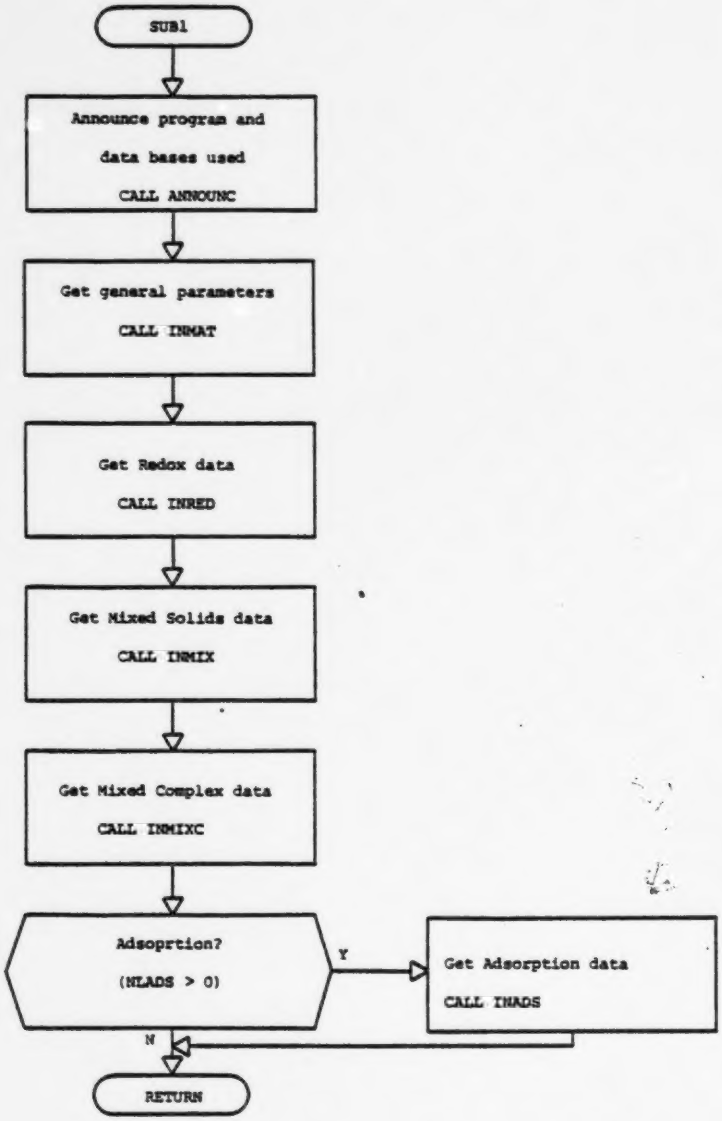
REDEQL-UMD MAIN PROGRAM

The MAIN routine is quite simple. It initializes the case counter to zero, calls SUB1 to perform input from the user deck and the Thermodynamic Data File, calls SUB2 to echo user input data, initial thermodynamic data, and (if requested) the thermodynamic data after correction to a fixed ionic strength. SUB3 is then called once per case to perform the computations for each case and to report output.


MAIN
CALL SUB1
CALL SUB2
Process cases
N = 0
Next case
N = N + 1
CALL SUB3
N
Last case?
(N = NCASES)
Y
STOP

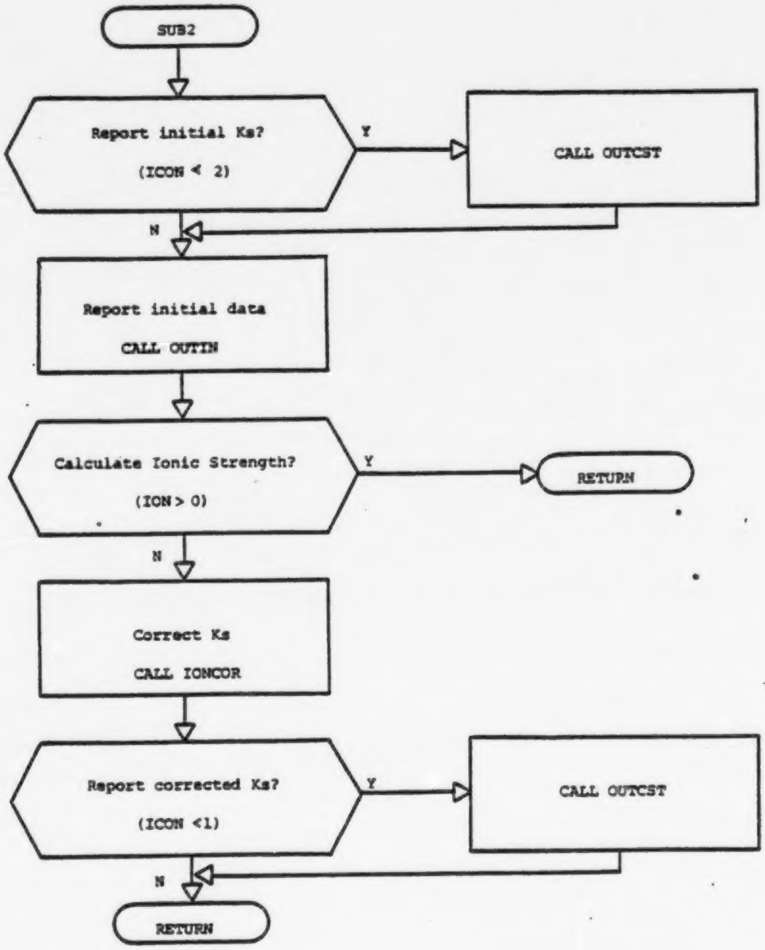
SUB1

SUB1 is the driver for input to REDEQL-UMD. It is called by the MAIN program. SUB1 calls ANOUNC to print the program header, INMAT to perform input of general parameters, metal and ligand concentrations, etc., INRED for input of redox data, INMIX for input of mixed solid data, INMIXC for input of mixed complex data, and (if adsorption is being considered) INADS for input of adsorption data.


SUB1
Announce program and
data bases used
CALL ANNOUNC
Get general parameters
CALL INMAT
Get Redox data
CALL INRED
Get Mixed Solids data
CALL INMIX
Get Mixed Complex data
CALL INMIXC
Adsoprtion?
(NLADS > 0)
Y
Get Adsorption data
CALL INADS
N
RETURN

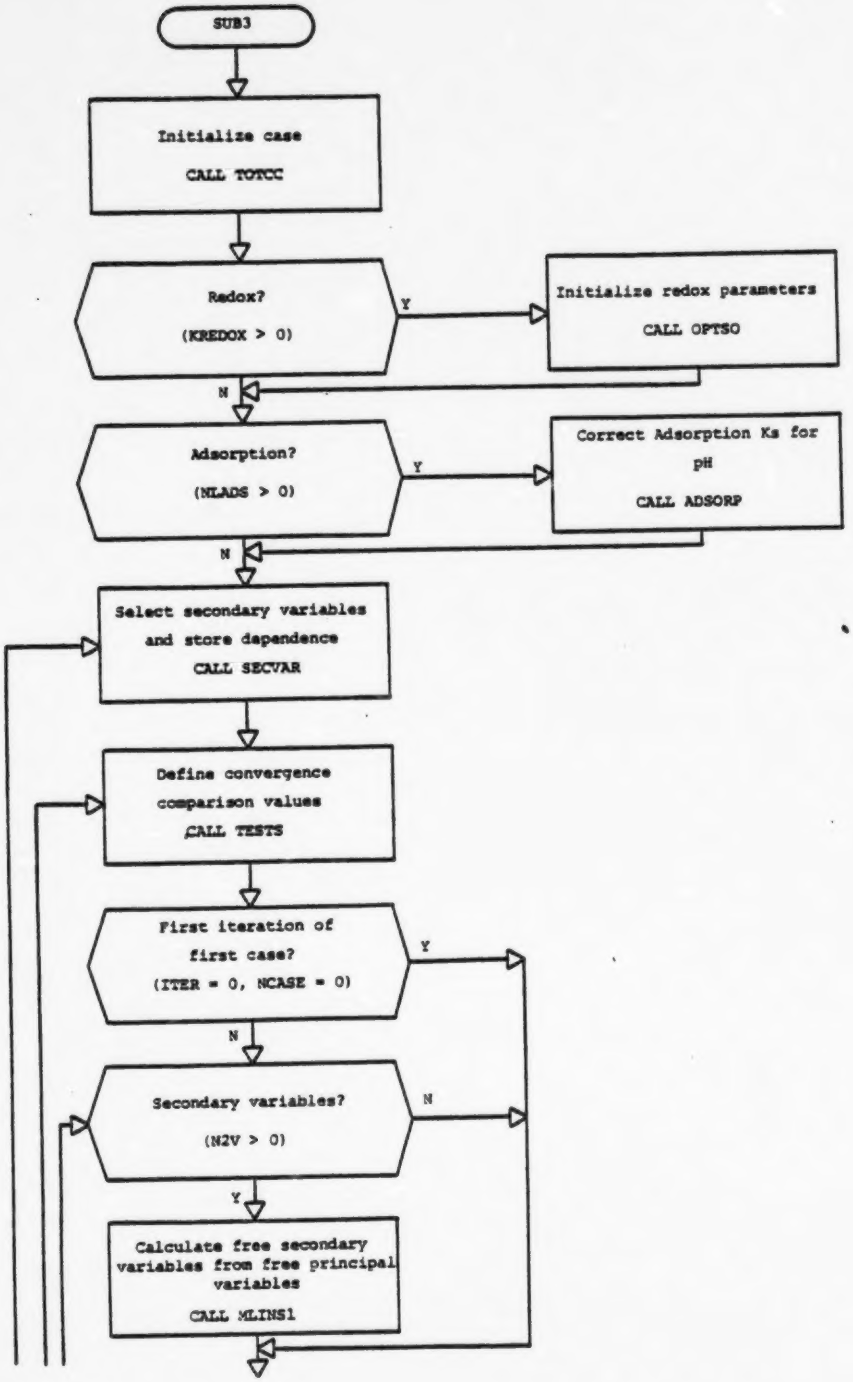
SUB2

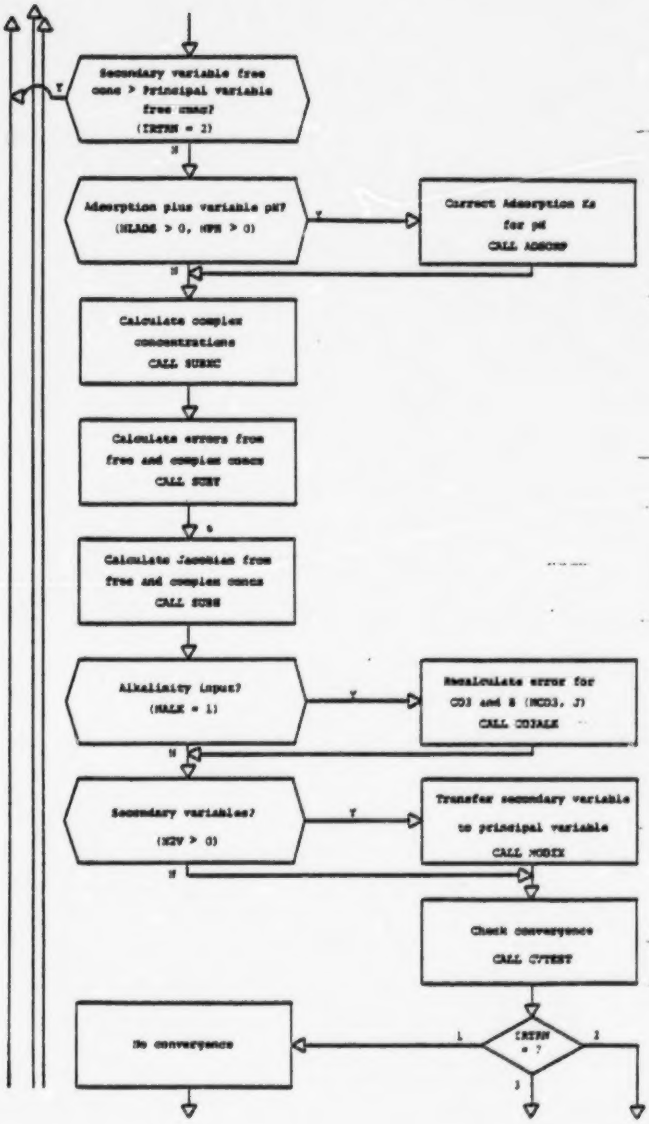
SUB2 reports the initial thermodynamic data and user input. If ionic strength is fixed, the thermodynamic constants are corrected and, if the user has so requested, thermodynamic data are reported after this correction.

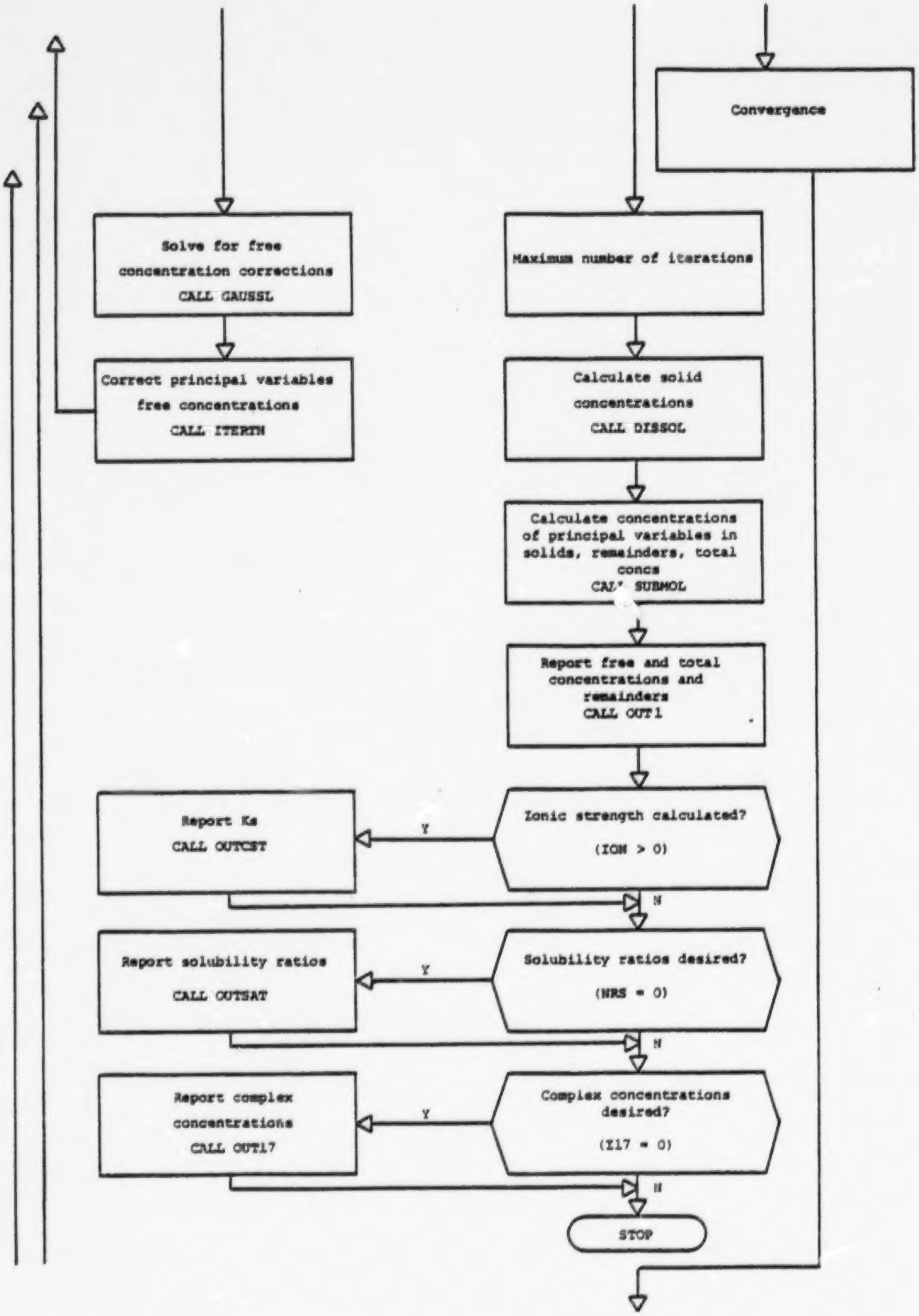

SUB2
Report initial Ks?
(ICON < 2)
Y
CALL OUTCST
N
Report initial data
CALL OUTIN
Calculate Ionic Strength?
(ION > 0)
Y
RETURN
N
Correct Ks
CALL IONCOR
Report corrected Ks?
(ICON <1)
Y
CALL OUTCST
N
RETURN

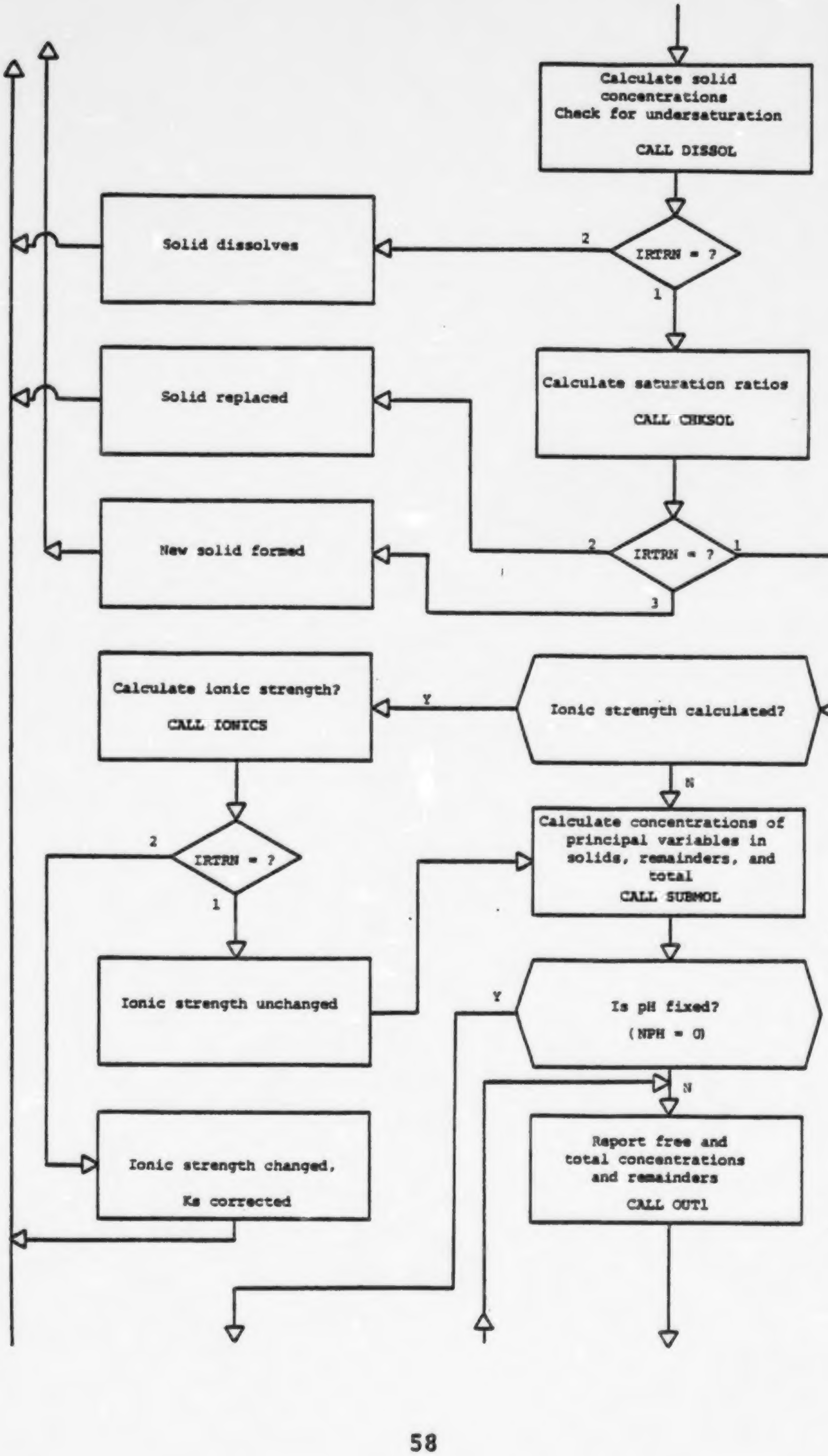
SUB3

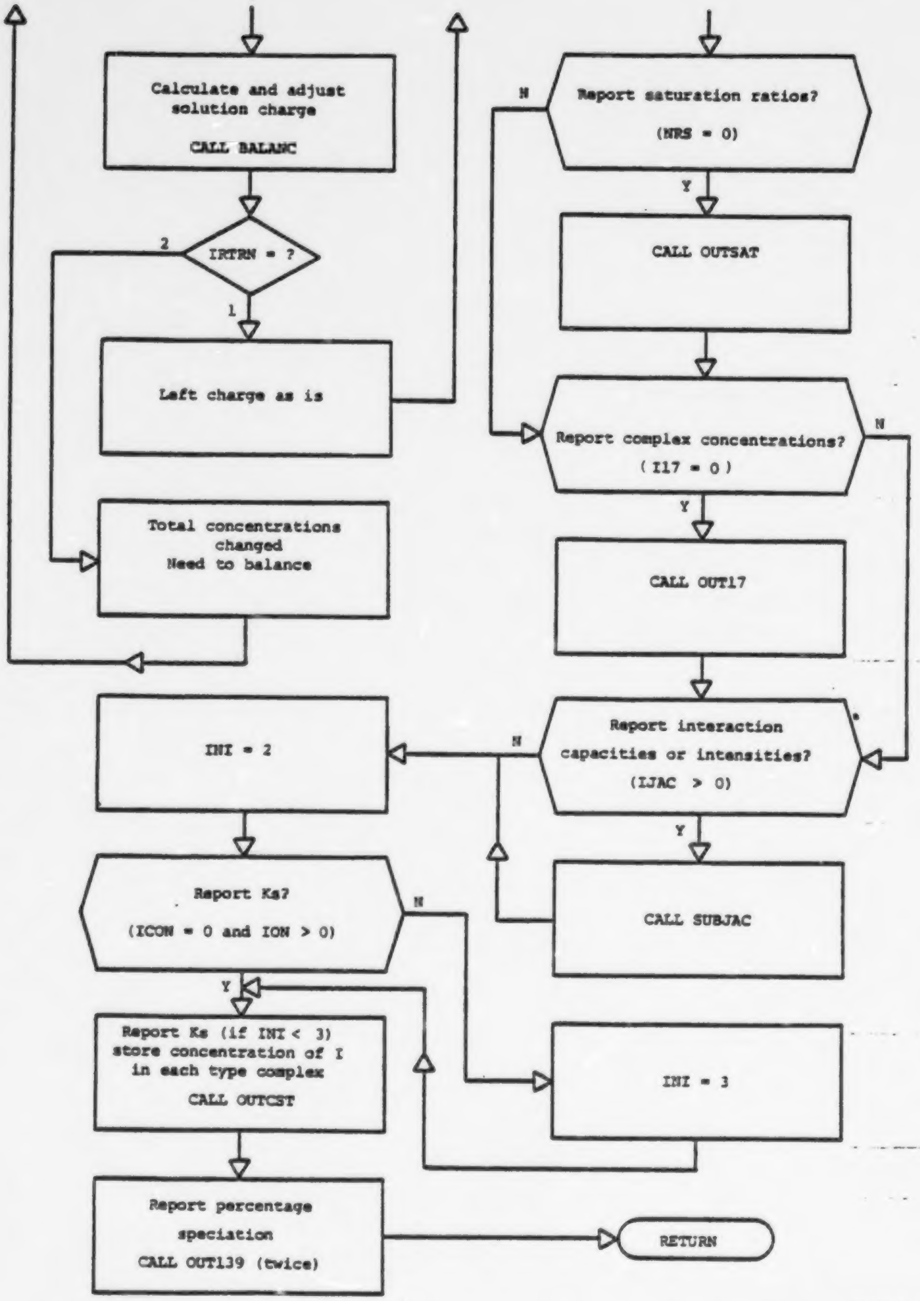
SUB3 is called by the MAIN program once per case to manage the computations for each case. SUB3 controls the overall flow of the computational portions of the program; thus an understanding of this routine is basic to an understanding of the heart of REDEQL-UMD.


SUB3
Initialize case
CALL TOTCC
Redox?
(KREDOX > 0)
Y
Initialize redox parameters
CALL OPTSO
N
Adsorption?
(NLADS > 0)
Y
Correct Adsorption Ks for pH
CALL ADSORP
N
Select secondary variables
and store dependence
CALL SECVAR
Define convergence
comparison values
CALL TESTS
First iteration of
first case?
(ITER = 0, NCASE = 0)
Y
N
Secondary variables?
(N2V > 0)
N
Y
Calculate free secondary
variables from free principal
variables
CALL MLINS1


Secondary variable free conc > Principal variable free conc?
(IRTRN = 2)
Y
N
Adsorption plus variable pH?
(NLADS > 0, NPH > 0)
Y
Correct Adsorption Ks for pH
CALL ADSORP
N
Calculate complex concentrations
CALL SUBMC
Calculate errors from free and complex concs
CALL SUBY
6
Calculate Jacobian from free and complex concs
CALL SUBB
Alkalinity input?
(NALK = 1)
Y
Recalculate error for CO3 and H (NCO3, J)
CALL CO3ALK
N
Secondary variables?
(N2V > 0)
Y
Transfer secondary variable to principal variable
CALL MODIX
N
Check convergence
CALL CVTEST
IRTRN = ?
1
2
3
No convergence

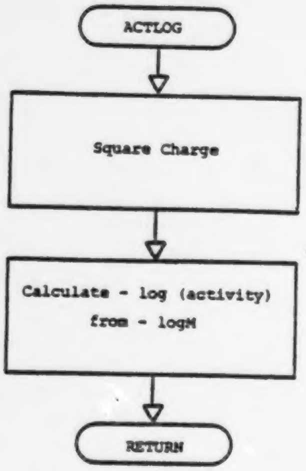

Convergence
Solve for free concentration corrections CALL GAUSSL
Maximum number of iterations
Correct principal variables free concentrations CALL ITERTH
Calculate solid concentrations CALL DISSOL
Calculate concentrations of principal variables in solids, remainders, total concs CALL SUBMOL
Report free and total concentrations and remainders CALL OUT1
Report Ks CALL OUTCST
Y
Ionic strength calculated? (ION > 0)
N
Report solubility ratios CALL OUTSAT
Y
Solubility ratios desired? (NRS = 0)
N
Report complex concentrations CALL OUT17
Y
Complex concentrations desired? (I17 = 0)
N
STOP


Calculate solid concentrations
Check for undersaturation
CALL DISSOL
IRTRN = ?
2
1
Solid dissolves
Calculate saturation ratios
CALL CHKSOL
Solid replaced
IRTRN = ?
2
1
3
New solid formed
Calculate ionic strength?
CALL IONICS
Y
Ionic strength calculated?
N
IRTRN = ?
2
1
Calculate concentrations of principal variables in solids, remainders, and total
CALL SUBMOL
Ionic strength unchanged
Y
Is pH fixed?
(NPH = 0)
N
Ionic strength changed,
Ks corrected
Report free and total concentrations and remainders
CALL OUT1


Calculate and adjust solution charge
CALL BALANC
IRTRN = ?
2
1
Left charge as is
Total concentrations changed
Need to balance
N
Report saturation ratios?
(NRS = 0)
Y
CALL OUTSAT
Report complex concentrations?
(I17 = 0)
N
Y
CALL OUT17
INI = 2
N
Report interaction capacities or intensities?
(IJAC > 0)
Y
CALL SUBJAC
Report Ks?
(ICON = 0 and ION > 0)
N
Y
Report Ks (if INI < 3)
store concentration of I
in each type complex
CALL OUTCST
INI = 3
Report percentage speciation
CALL OUT139 (twice)
RETURN

ACTLOG

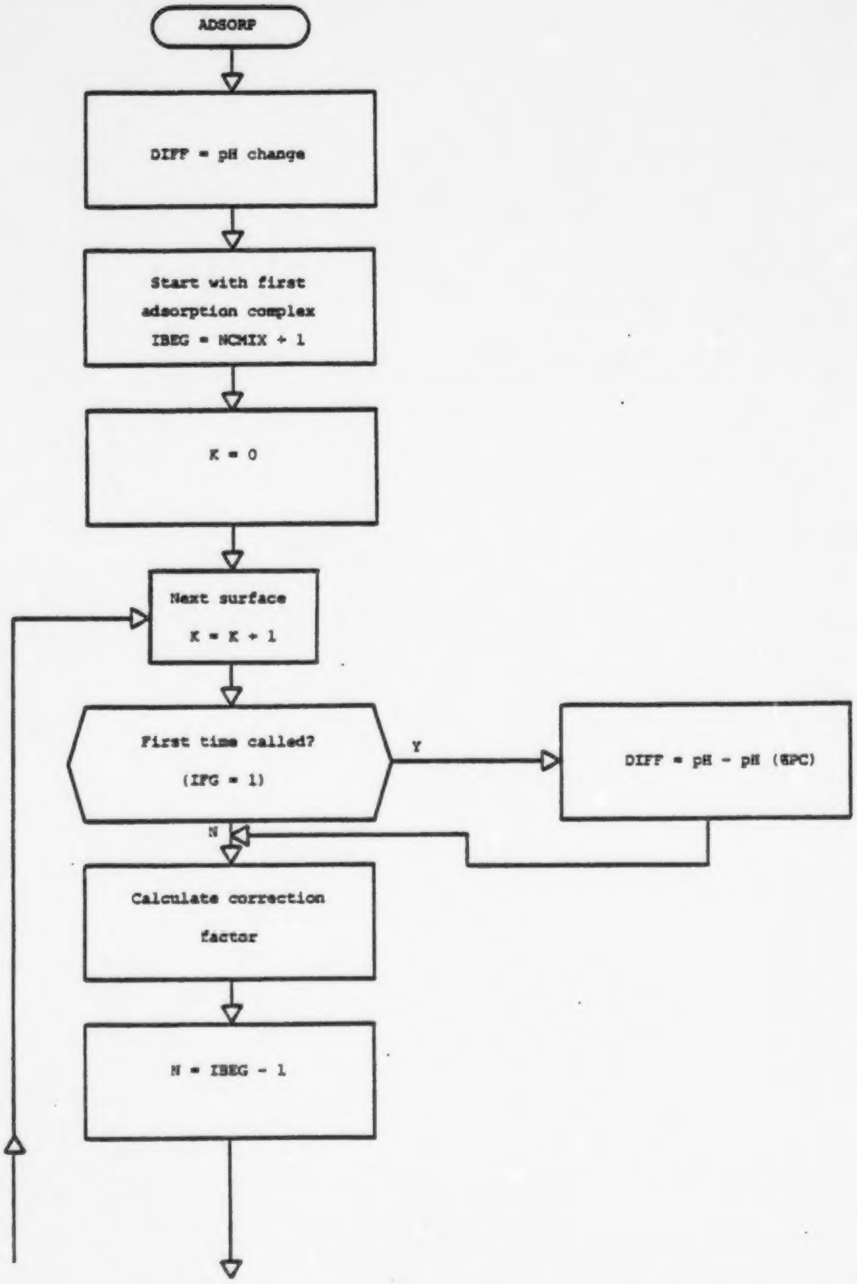
Function ACTLOG is used by routines OUT1 and OUT17 to calculate the negative log of the activity of an ion from the concentration and the charge of the ion (parameters CLOG and ICHRG respectively). The calculation is performed via the Magnuson or Davies equation with constants previously assigned in IONCOR.

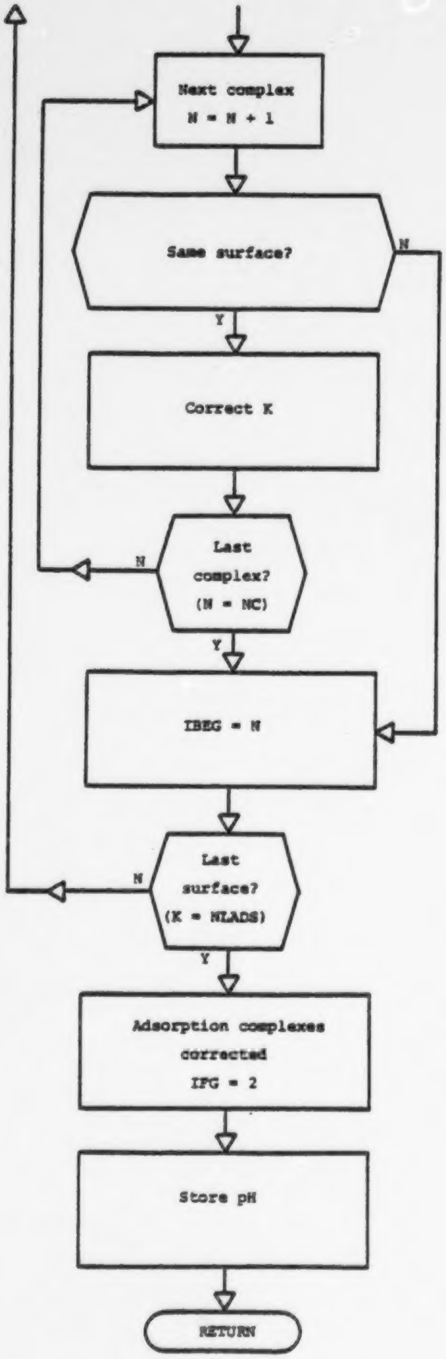

ACTLOG
Square Charge
Calculate - log (activity)
from - logM
RETURN

ADSORP

ADSORP is used by SUB3 to perform corrections for pH on the "apparent" formation constants of "adsorption complexes".

The apparent K equals the intrinsic constant, KA, multiplied by 10 to the power alpha * (pH - pH(ZPC))*(Q(M)-n) where Q(M) is the metal charge and n is the number of H replaced on the surface.


ADSORP
DIFF = pH change
Start with first adsorption complex IBEG = NCMIX + 1
K = 0
Next surface K = K + 1
First time called? (IFG = 1)
Y
DIFF = pH - pH (ZPC)
N
Calculate correction factor
N = IBEG - 1


Next complex
N = N + 1
Same surface?
N
Y
Correct K
Last
complex?
(N = NC)
N
Y
IBEG = N
Last
surface?
(K = NLADS)
N
Y
Adsorption complexes
corrected
IFG = 2
Store pH
RETURN

ALLOTC

Subroutine ALLOTC is called by the input routines INADS, INMIXC, and INMAT to store data for complexes. ALLOTC is called with MI (metal number), LI (ligand number), either LI or MI (third ion) in mixed complexes, and variables FRST and LAST as parameters. FRST and LAST indicate the range of the current data line for which data is to be stored.

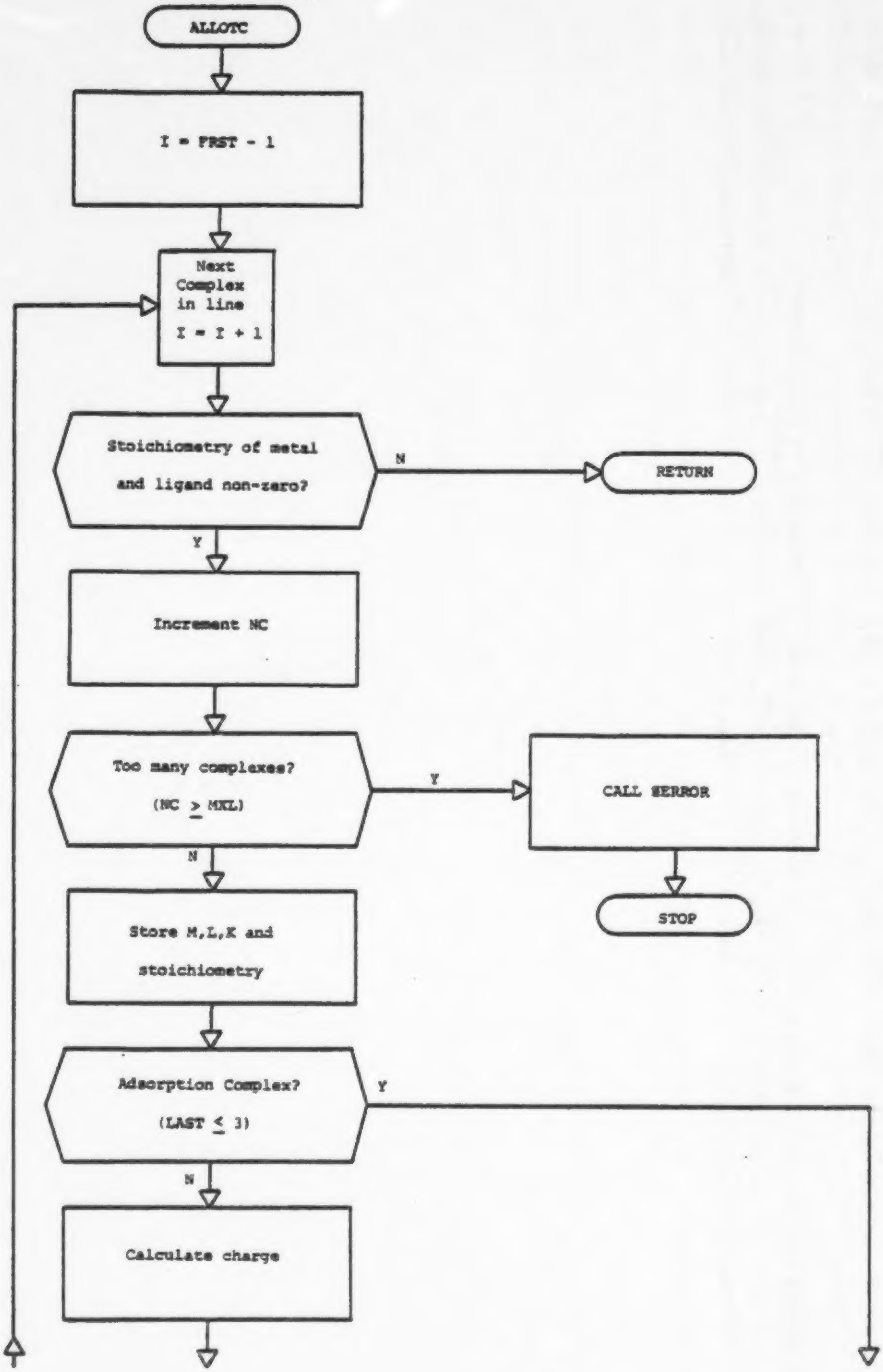
If no overflow occurs, the data for each complex are stored as follows:

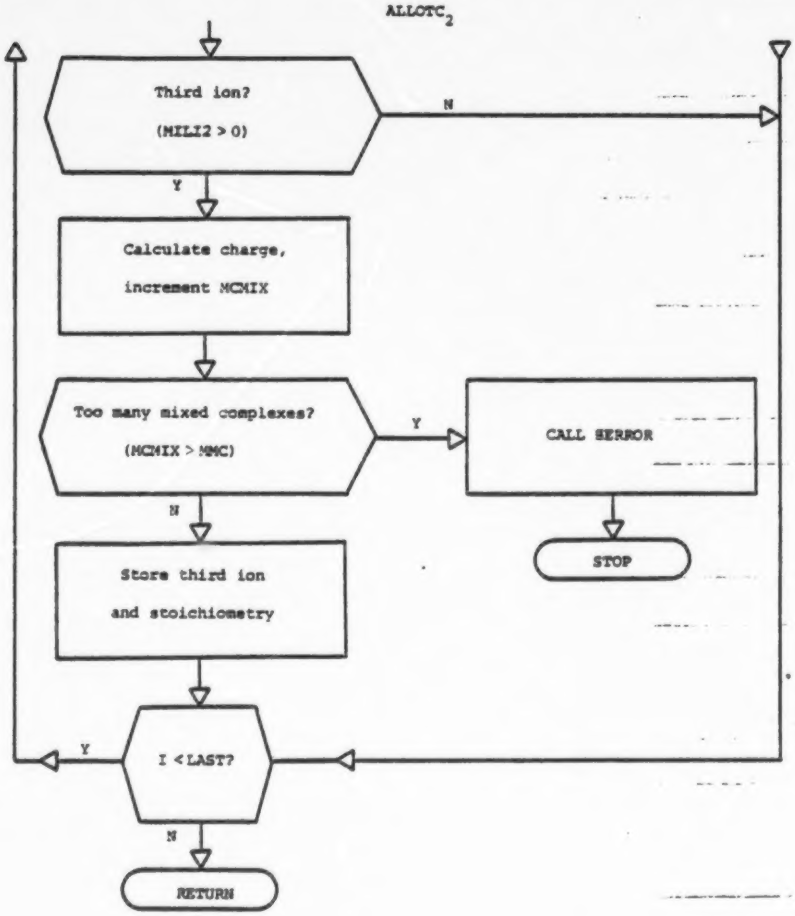
CMET(NC)	metal variable number
CLIG(NC)	ligand variable number
CM(NC)	metal stoichiometric coefficient
CL(NC)	ligand stoichiometric coefficient
CHOH(NC)	hydrogen or hydroxide stoichiometric coefficient
CHR(NC)	total charge on the complex (not for adsorption)
CK(NC)	stability constant for the complex (-logK)
CLIG2(MCMIX)	third ion variable number (mixed complexes only)
CL2(MCMIX)	third ion stoichiometric coefficient (mixed complexes)

where:

NC = complex count (i.e. position of the complex being stored)

MCMIX = mixed complex count


ALLOTC
I = FRST - 1
Next Complex in line
I = I + 1
Stoichiometry of metal and ligand non-zero?
N
RETURN
Y
Increment NC
Too many complexes? (NC ≥ MXL)
Y
CALL SERROR
STOP
N
Store M,L,K and stoichiometry
Adsorption Complex? (LAST ≤ 3)
Y
N
Calculate charge

ALLOTC$_2$
Third ion?
(MILI2 > 0)
N
Y
Calculate charge,
increment MCMIX
Too many mixed complexes?
(MCMIX > MMC)
Y
CALL SERROR
STOP
N
Store third ion
and stoichiometry
Y
I < LAST?
N
RETURN

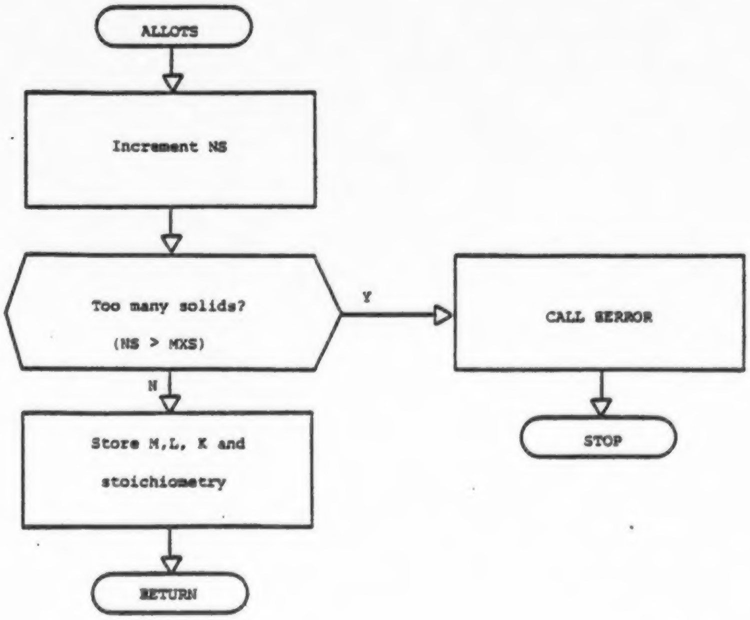
ALLOTS

Subroutine ALLOTS is called by INMAT to store data for solids in a manner similar to that used by ALLOTC for storing data for complexes. ALLOTS is called with MI (metal number), LI (ligand number), and NUM (position in current data line). Unlike ALLOTC however, ALLOTS stores data for only one solid each time it is called.

Solid data are stored as follows:

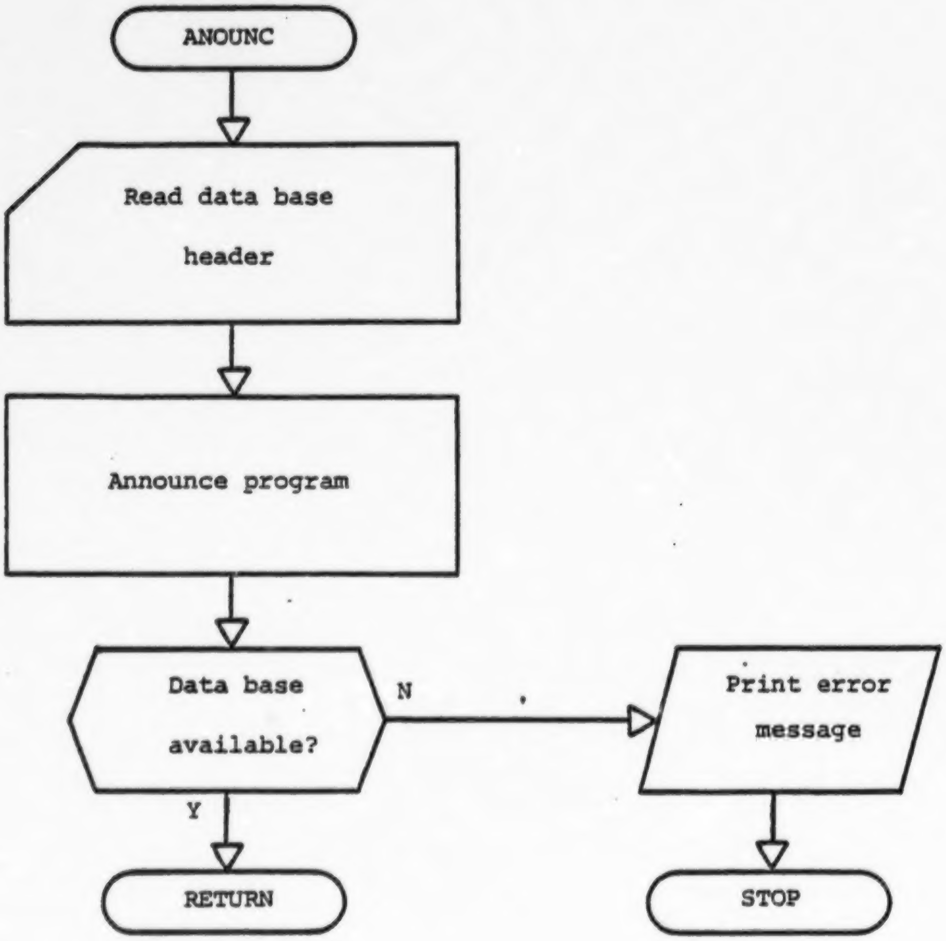
KMET(NS)	metal variable number
KLIG(NS)	ligand variable number
KSM(NS)	metal stoichiometric coefficient
KSL(NS)	ligand stoichiometric coefficient
KSHOH(NS)	H+ or OH- stoichiometric coefficient
SOLHY(NS)	formation constant for the solid (-log)

where NS = solid count.


ALLOTS
Increment NS
Too many solids?
(NS > MXS)
Y
N
CALL SERROR
STOP
Store M,L, K and
stoichiometry
RETURN

ANOUNC

Subroutine ANOUNC is called by SUB1 at the beginning of program execution to print header data which includes the program name, the current date and time, the date recorded in the first line of the thermodynamic data base being used for the run, and the temperature for which the thermodynamic data base was compiled. If an "end of file" condition is encountered upon attempting to read the data base, the run is aborted with an informative error message.

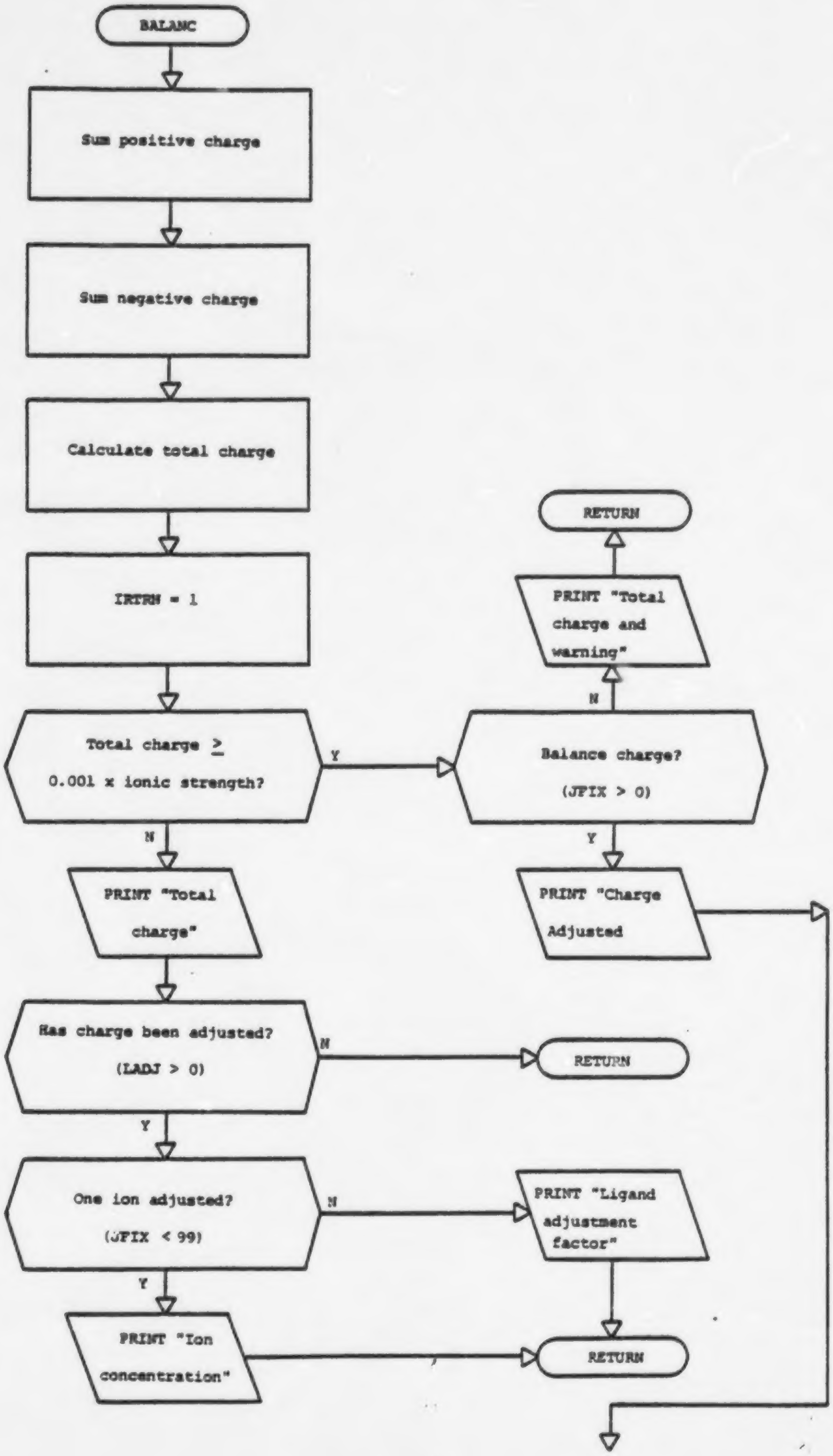

ANOUNC
Read data base header
Announce program
Data base available?
N
Y
Print error message
RETURN
STOP

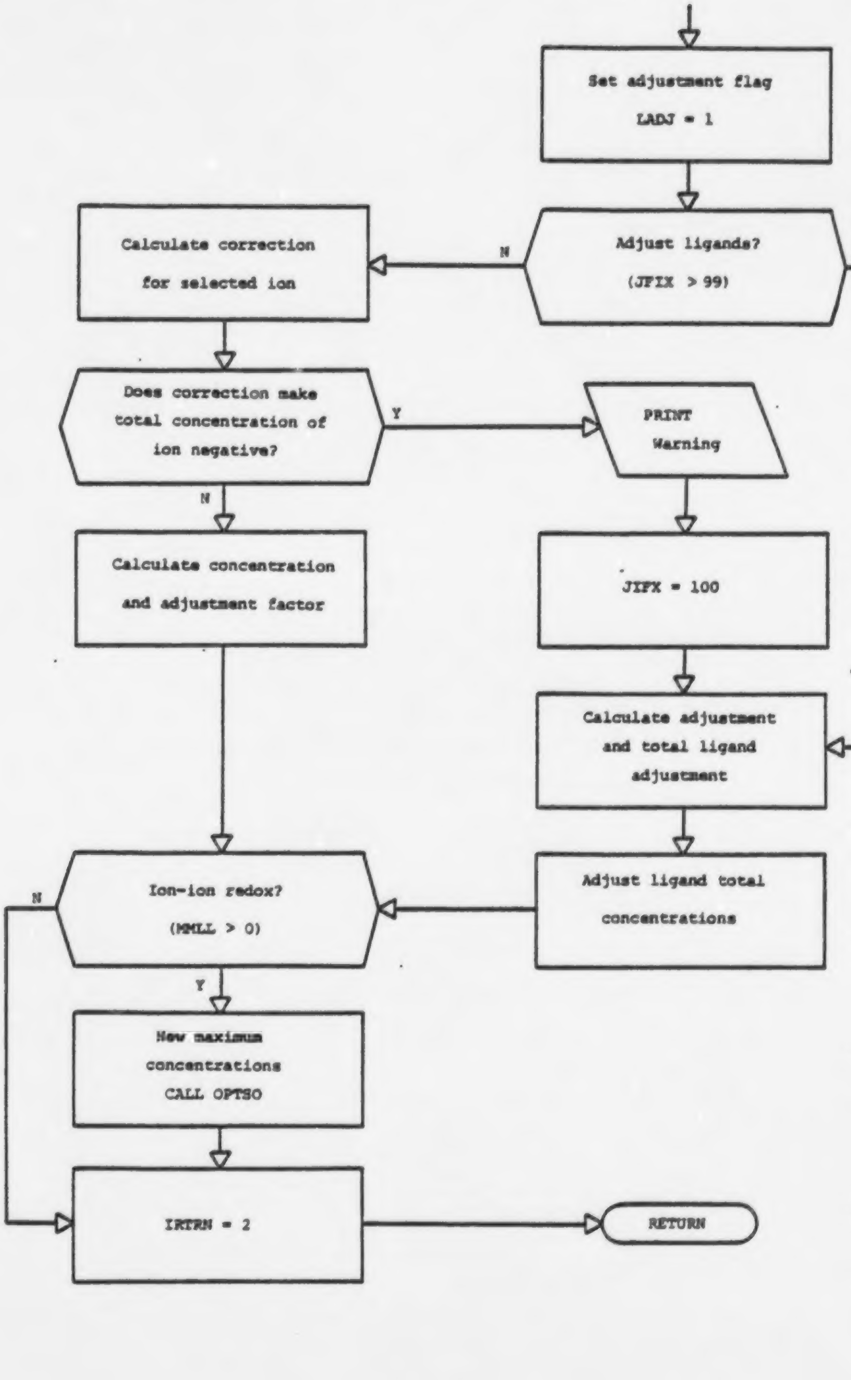
BALANC

Subroutine BALANC is called by SUB3, if pH is fixed, when all convergence criteria of the program have been satisfied, in order to compute the total charge on the solution being modeled, to adjust the concentration(s) of one ion or all ligands (if the user has so specified), and to report the charge on the solution when no further changes are to be made.

The total charge is calculated according to:

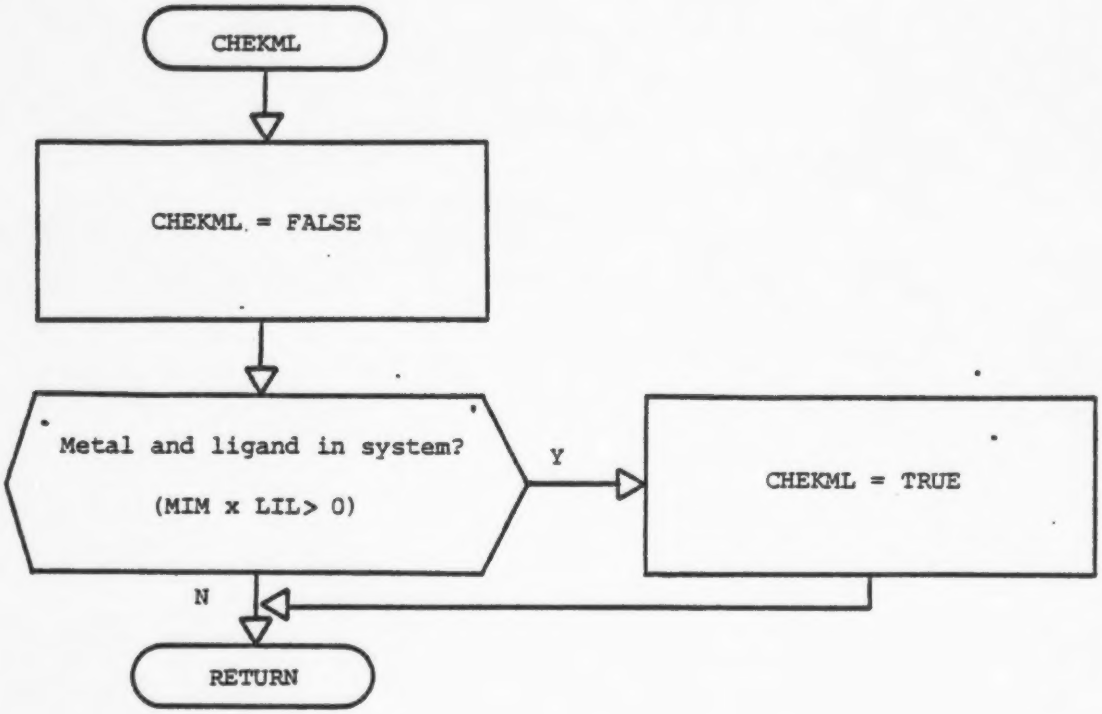
$$\text{TOTAL CHARGE} = \text{TOTH} + \sum\left[\text{METAL TOTAL CONC.} \times \text{METAL CHARGE}\right] + \sum\left[\text{LIGAND TOTAL CONC.} \times \text{LIGAND CHARGE}\right] - \text{ELECTRONS IN SOLIDS AND COMPLEXES.}$$


BALANC
Sum positive charge
Sum negative charge
Calculate total charge
IRTRN = 1
Total charge ≥ 0.001 x ionic strength?
Y
N
PRINT "Total charge"
Has charge been adjusted? (LADJ > 0)
N
RETURN
Y
One ion adjusted? (JFIX < 99)
N
PRINT "Ligand adjustment factor"
Y
PRINT "Ion concentration"
RETURN
Balance charge? (JFIX > 0)
N
PRINT "Total charge and warning"
RETURN
Y
PRINT "Charge Adjusted


Set adjustment flag
LADJ = 1
Adjust ligands?
(JFIX > 99)
N
Y
Calculate correction
for selected ion
Does correction make
total concentration of
ion negative?
Y
N
PRINT
Warning
Calculate concentration
and adjustment factor
JIFX = 100
Calculate adjustment
and total ligand
adjustment
Adjust ligand total
concentrations
Ion-ion redox?
(MMLL > 0)
N
Y
New maximum
concentrations
CALL OPTSO
IRTRN = 2
RETURN

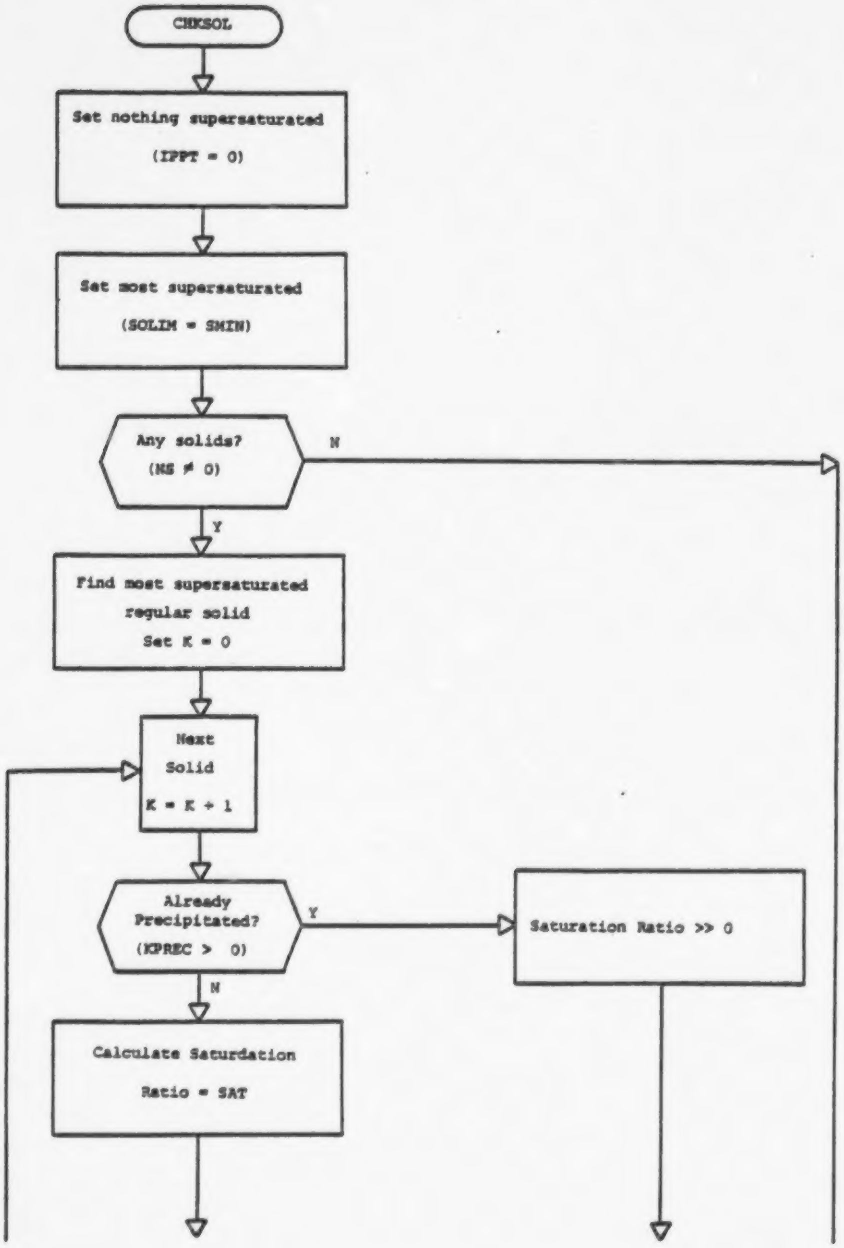
CHEKML

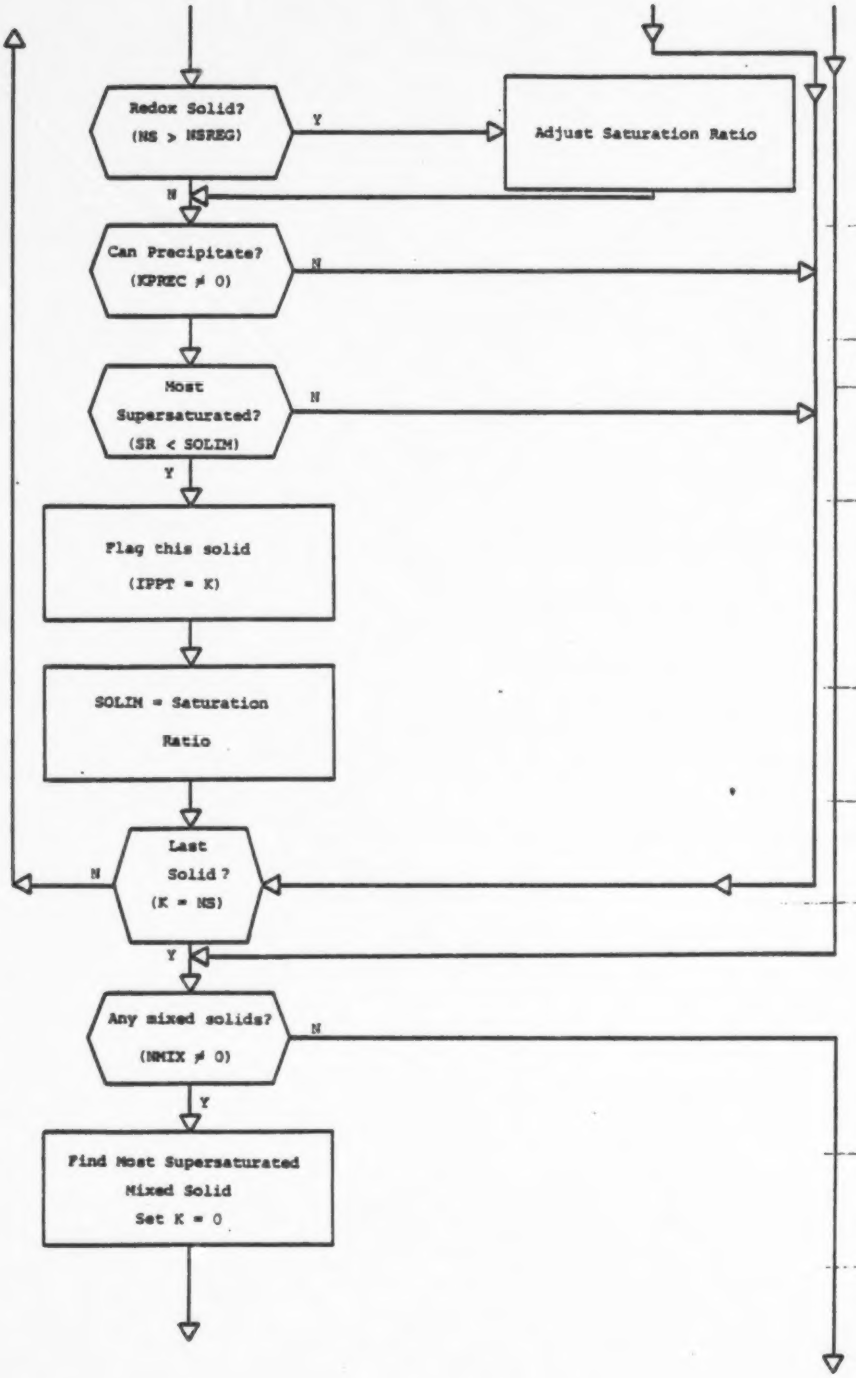
Logical function CHEKML is used to determine whether a metal-ligand pair can occur within the system defined by the user. CHEKML is called with the metal reference number and the ligand reference number as parameters and returns a value of TRUE if the M-L pair is possible. Additionally, the two parameters are set to the appropriate MI and LI pointers (variable numbers) for the metal and ligand.

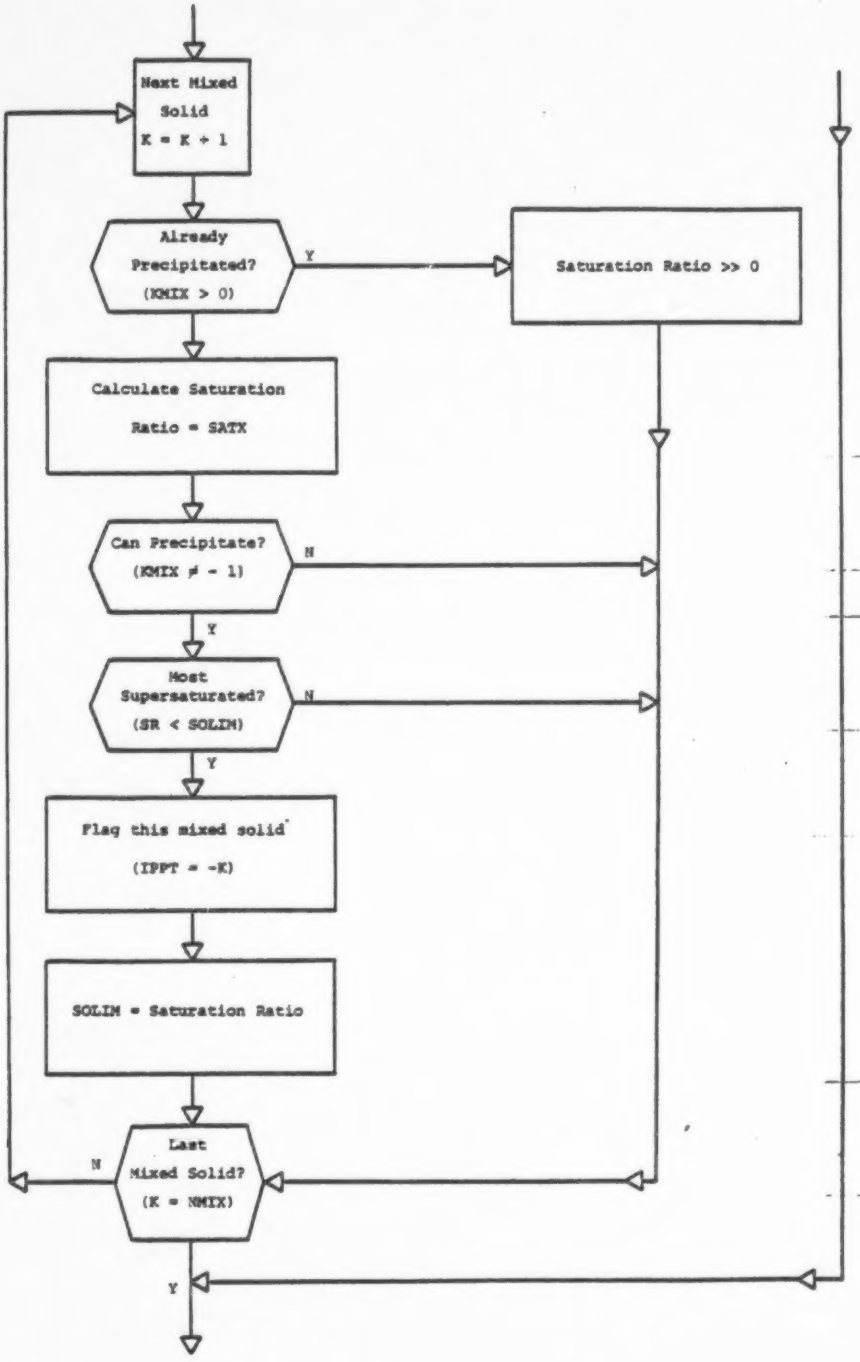

CHEKML
CHEKML = FALSE
Metal and ligand in system?
(MIM x LIL> 0)
Y
CHEKML = TRUE
N
RETURN

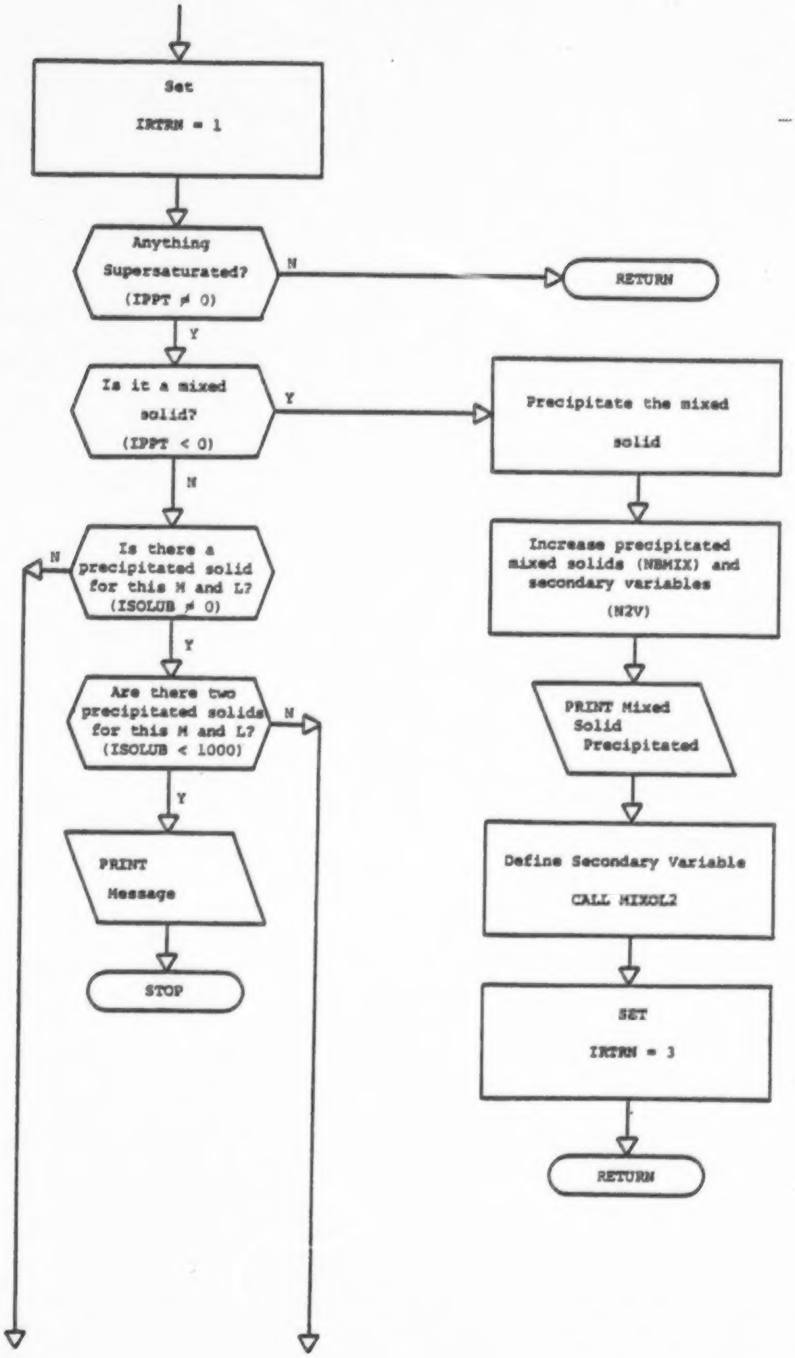
CHKSOL

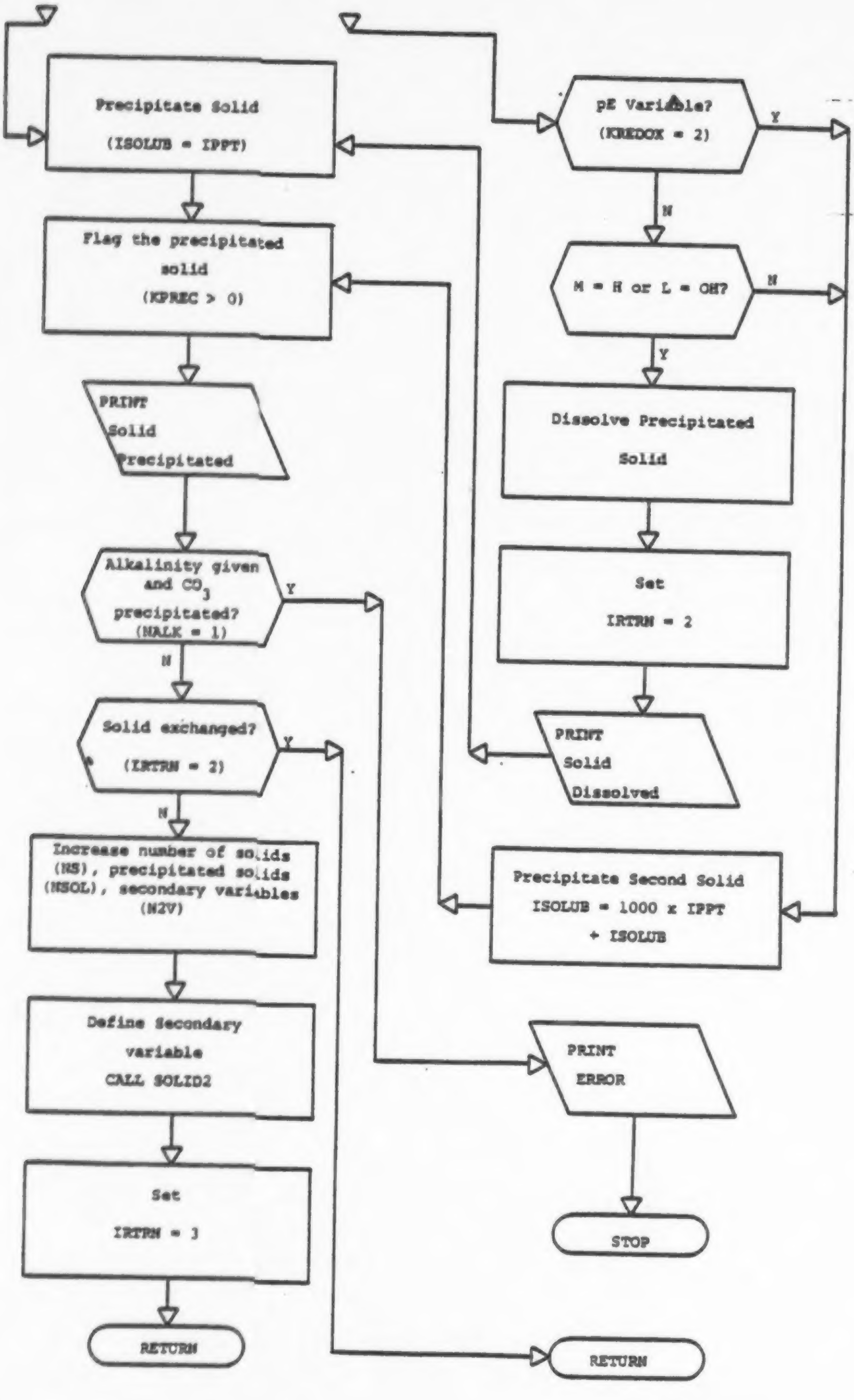
Subroutine CHKSOL is called by SUB3 to calculate the degrees of saturation of "regular" solids and mixed solids, to find the one which is most supersaturated (if any), and to precipitate it. If no solid or mixed solid is supersaturated, no action is taken. If a solid of the same M-L pair as the most supersaturated "regular" solid has already been precipitated, that solid may be "dissolved" prior to precipitation of the new solid. Degrees of saturation (-log) are stored in SAT and SATX for solids and mixed solids respectively.


CHKSOL
Set nothing supersaturated
(IPPT = 0)
Set most supersaturated
(SOLIM = SMIN)
Any solids?
(NS ≠ 0)
N
Y
Find most supersaturated
regular solid
Set K = 0
Next
Solid
K = K + 1
Already
Precipitated?
(KPREC > 0)
Y
Saturation Ratio >> 0
N
Calculate Saturdation
Ratio = SAT


Redox Solid?
(NS > NSREG)
Y
Adjust Saturation Ratio
N
Can Precipitate?
(KPREC ≠ 0)
N
Most
Supersaturated?
(SR < SOLIM)
N
Y
Flag this solid
(IPPT = K)
SOLIM = Saturation
Ratio
N
Last
Solid ?
(K = NS)
Y
Any mixed solids?
(NMIX ≠ 0)
N
Y
Find Most Supersaturated
Mixed Solid
Set K = 0

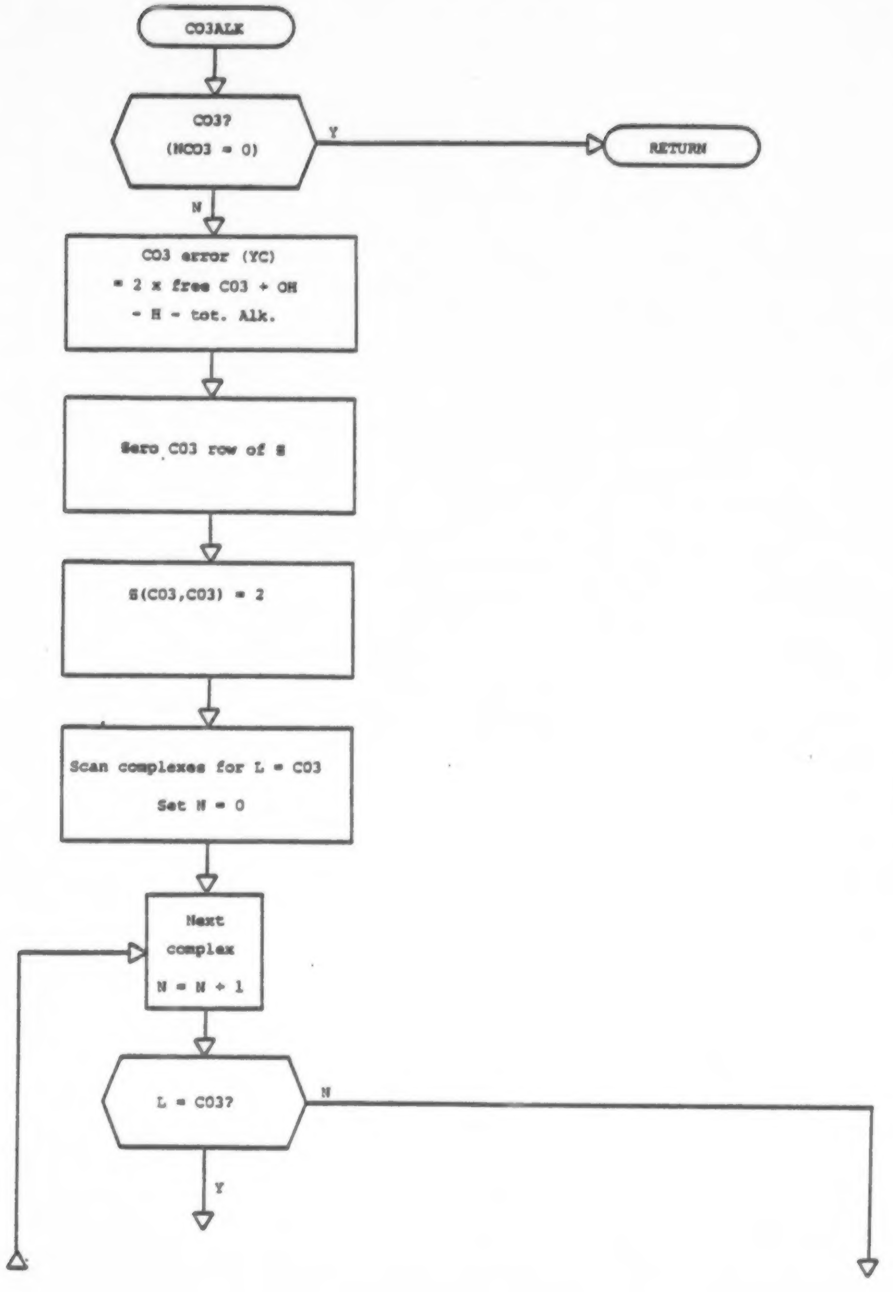

Next Mixed Solid K = K + 1
Already Precipitated? (KMIX > 0)
Y
Saturation Ratio >> 0
Calculate Saturation Ratio = SATX
Can Precipitate? (KMIX ≠ - 1)
N
Y
Most Supersaturated? (SR < SOLIM)
N
Y
Flag this mixed solid (IPPT = -K)
SOLIM = Saturation Ratio
Last Mixed Solid? (K = NMIX)
N
Y

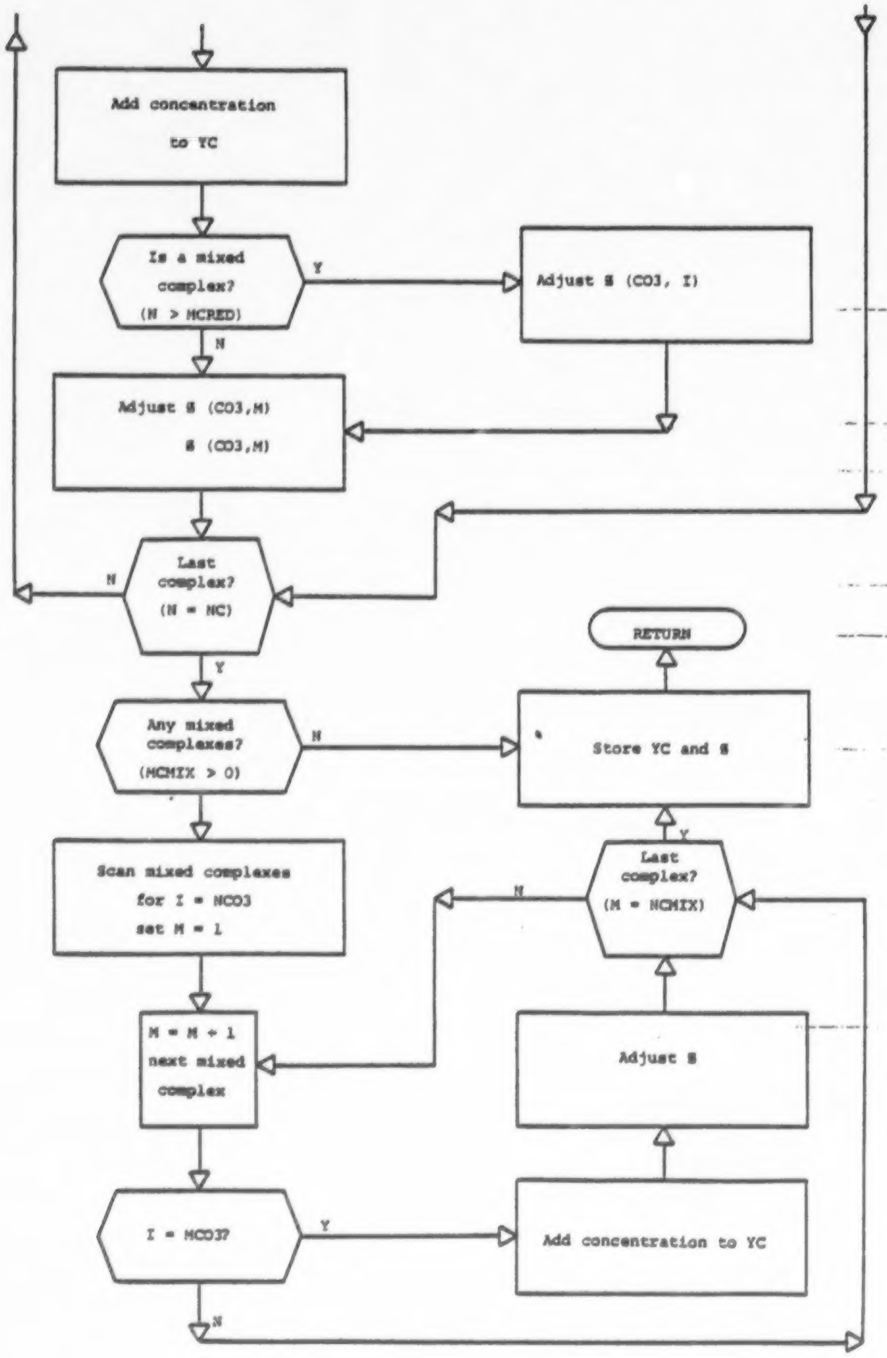

Set
IRTRN = 1
Anything
Supersaturated?
(IPPT ≠ 0)
N
RETURN
Y
Is it a mixed
solid?
(IPPT < 0)
Y
Precipitate the mixed
solid
N
Increase precipitated
mixed solids (NBMIX) and
secondary variables
(N2V)
N
Is there a
precipitated solid
for this M and L?
(ISOLUB ≠ 0)
Y
PRINT Mixed
Solid
Precipitated
Are there two
precipitated solids
for this M and L?
(ISOLUB < 1000)
N
Y
Define Secondary Variable
CALL MIXOL2
PRINT
Message
STOP
SET
IRTRN = 3
RETURN


Precipitate Solid
(ISOLUB = IPPT)
Flag the precipitated solid
(KPREC > 0)
PRINT
Solid
Precipitated
Alkalinity given and CO3 precipitated?
(NALK = 1)
Y
N
Solid exchanged?
(IRTRN = 2)
Y
N
Increase number of solids (NS), precipitated solids (NSOL), secondary variables (N2V)
Define Secondary variable
CALL SOLID2
Set
IRTRN = 3
RETURN
pE Variable?
(KREDOX = 2)
Y
N
M = H or L = OH?
N
Y
Dissolve Precipitated Solid
Set
IRTRN = 2
PRINT
Solid
Dissolved
Precipitate Second Solid
ISOLUB = 1000 x IPPT
+ ISOLUB
PRINT
ERROR
STOP
RETURN

CO3ALK

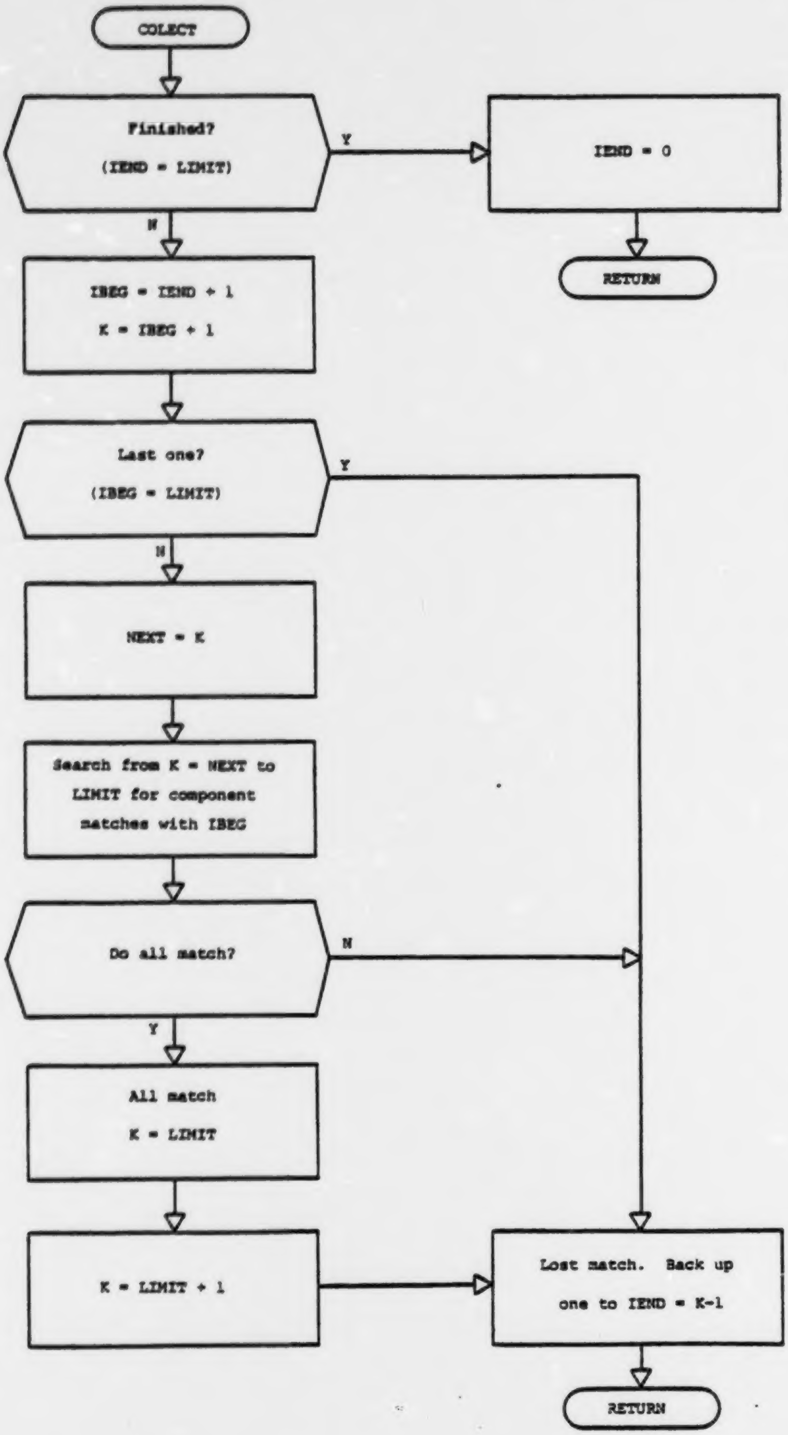
Subroutine CO3ALK calculates errors for CO3 and CO3 elements of the Z matrix when alkalinity is an input parameter.


CO3ALK
CO3?
(NCO3 = 0)
Y
RETURN
N
CO3 error (YC)
= 2 x free CO3 + OH
- H - tot. Alk.
Zero CO3 row of S
S(CO3,CO3) = 2
Scan complexes for L = CO3
Set N = 0
Next
complex
N = N + 1
L = CO3?
N
Y


Add concentration to YC
Is a mixed complex? (N > MCRED)
Y
N
Adjust $ (CO3, I)
Adjust $ (CO3,M)
$ (CO3,M)
Last complex? (N = NC)
N
Y
RETURN
Any mixed complexes? (MCMIX > 0)
N
Store YC and $
Scan mixed complexes for I = NCO3 set M = 1
M = M + 1 next mixed complex
Last complex? (M = NCMIX)
N
Y
Adjust $
I = NCO3?
Y
Add concentration to YC
N

COLECT

Subroutine COLECT is used by the output routines OUT17 and OUTCST to find the positions of the first (IBEG) and last (IEND) solid or complex of a particular M-L pair. IBEG and IEND are then used by the calling routine as loop parameters for printing output for that M-L pair. The variable LIMIT marks the position of the last possible complex or solid beyond which COLECT will not search; LIMIT is set by the calling routine.

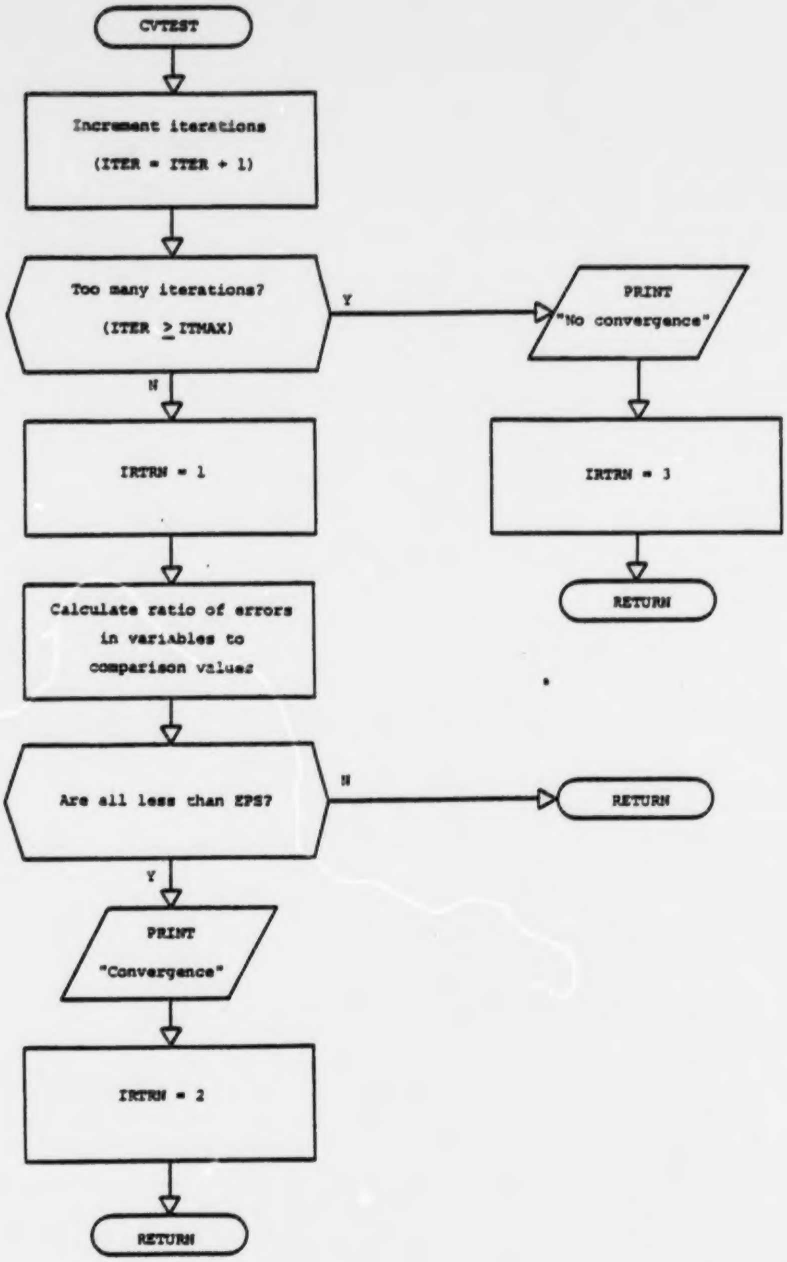

COLECT
Finished?
(IEND = LIMIT)
Y
IEND = 0
RETURN
N
IBEG = IEND + 1
K = IBEG + 1
Last one?
(IBEG = LIMIT)
Y
N
NEXT = K
Search from K = NEXT to
LIMIT for component
matches with IBEG
Do all match?
N
Y
All match
K = LIMIT
K = LIMIT + 1
Lost match. Back up
one to IEND = K-1
RETURN

CVTEST

The system being modeled is said to be convergent if for every variable, the error is insignificant with respect to the adjusted total concentration:

$$\mathrm{ABS}(Y(I) / \mathrm{TESTY}(I)) < \mathrm{EPS}$$

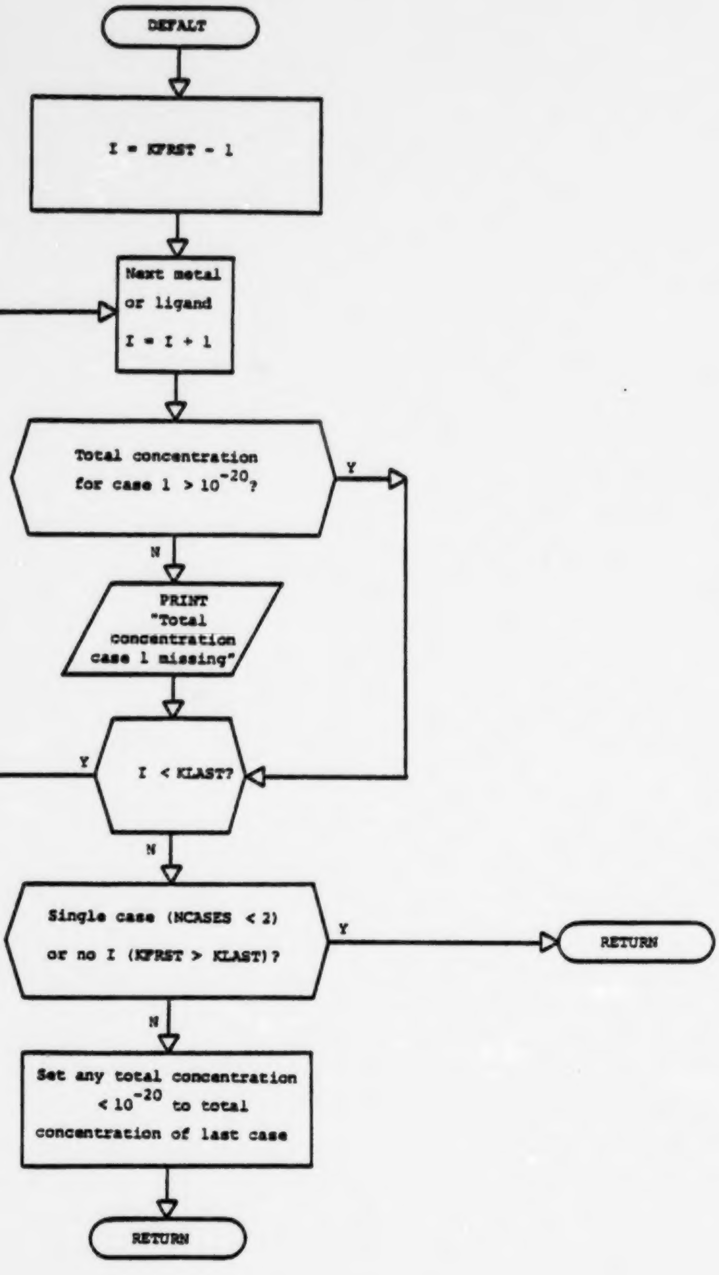
where Y(I) is the error for the variable, TESTY(I) is the comparison value from total concentrations, and EPS is a specified value of precision. Subroutine CVTEST performs this convergence test for the system.


CVTEST
Increment iterations
(ITER = ITER + 1)
Too many iterations?
(ITER $\geq$ ITMAX)
Y
PRINT
"No convergence"
N
IRTRN = 1
IRTRN = 3
RETURN
Calculate ratio of errors
in variables to
comparison values
Are all less than EPS?
N
RETURN
Y
PRINT
"Convergence"
IRTRN = 2
RETURN

DEFALT

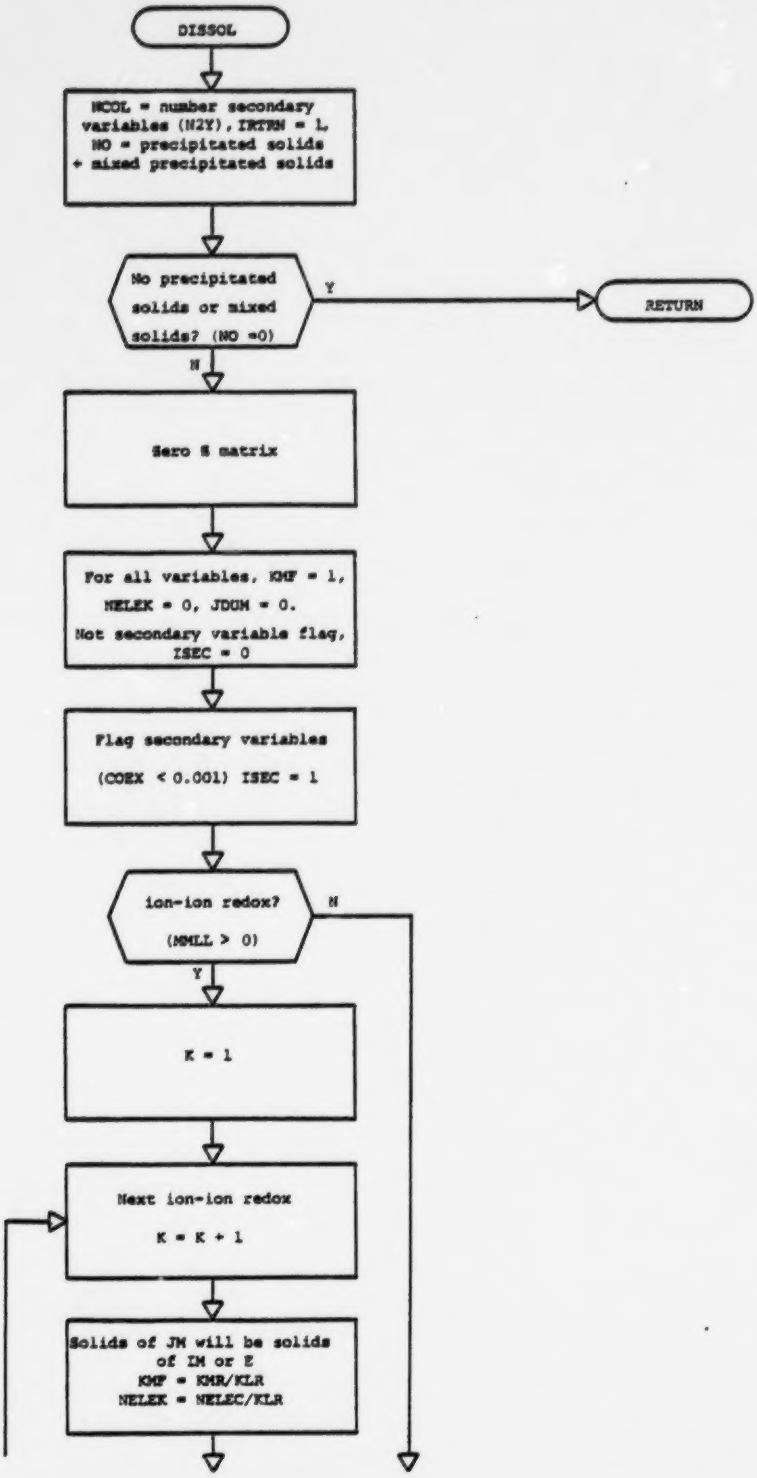
As described in the user's guide for REDEQL-UMD, when calculations involve more than one case, the total concentrations of metals and ligands need be specified only for the first case; the value for the previous case will be used by the program whenever a total concentration is not given for subsequent cases. Subroutine DEFALT implements this feature.

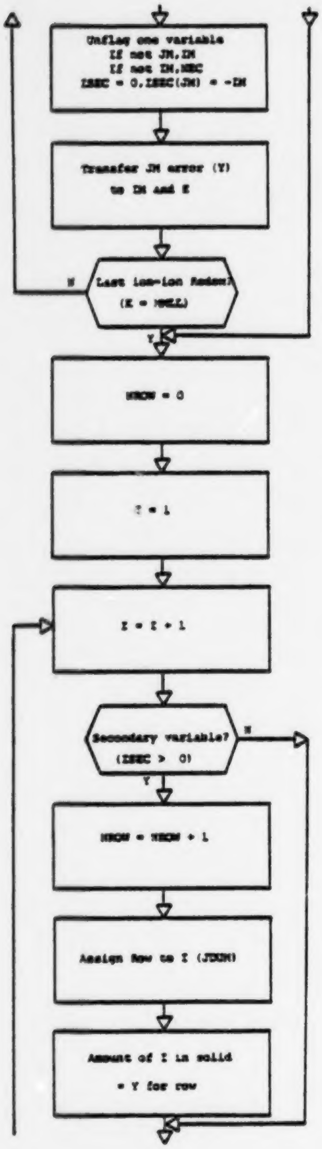
DEFALT is called with either TMIN (if concentrations are being input as -log M) or TMINW (if concentrations are being input as mg/liter) and the positions of the first (KFRST) and (KLAST) metal and/or ligand as parameters.

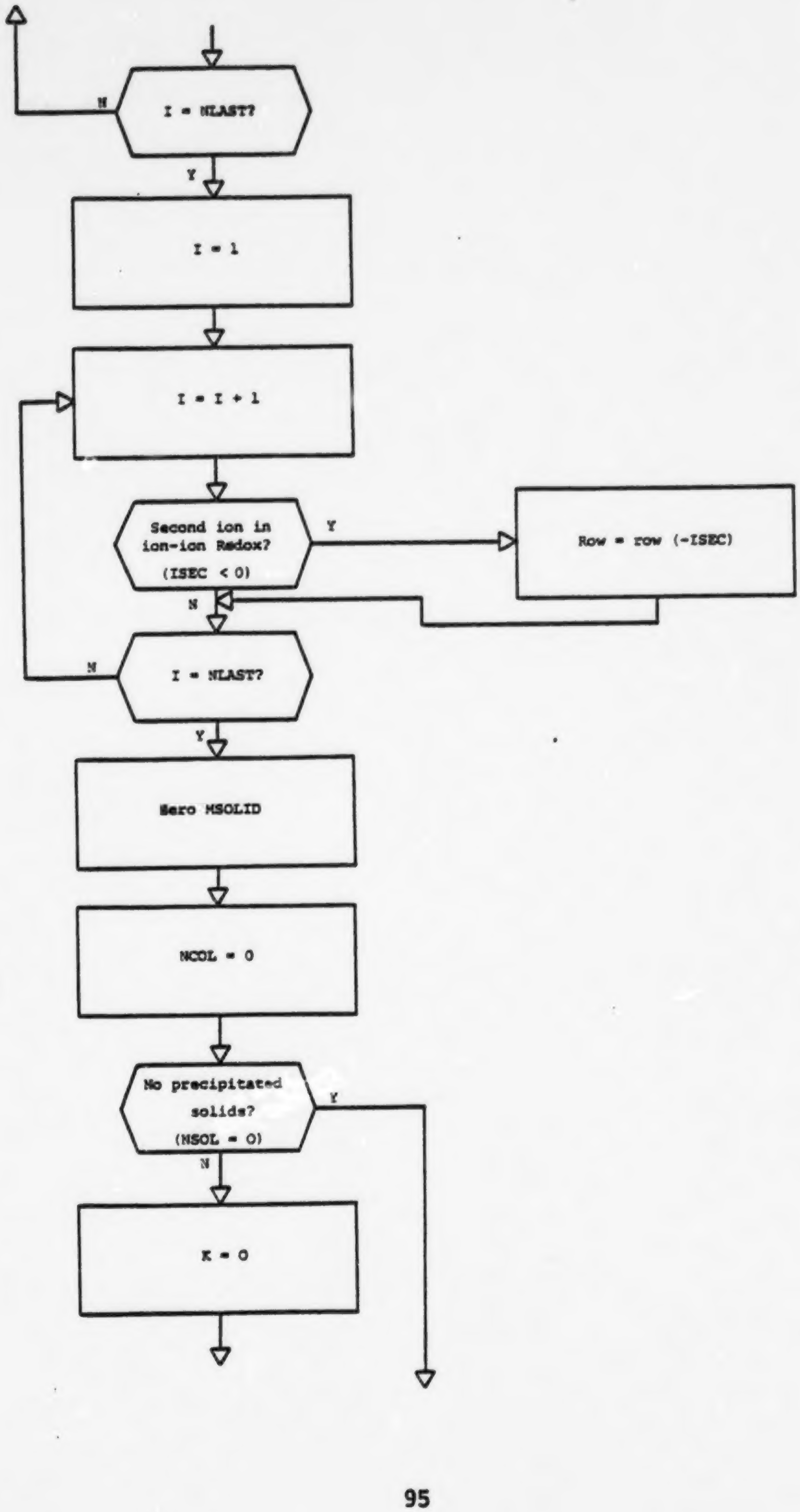

DEFALT
I = KFRST - 1
Next metal or ligand I = I + 1
Total concentration for case 1 > 10^{-20}?
Y
N
PRINT "Total concentration case 1 missing"
I < KLAST?
Y
N
Single case (NCASES < 2) or no I (KFRST > KLAST)?
Y
RETURN
N
Set any total concentration < 10^{-20} to total concentration of last case
RETURN

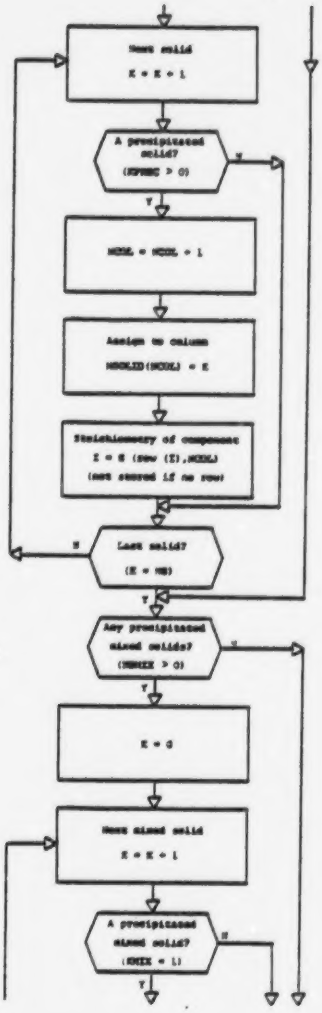
DISSOL

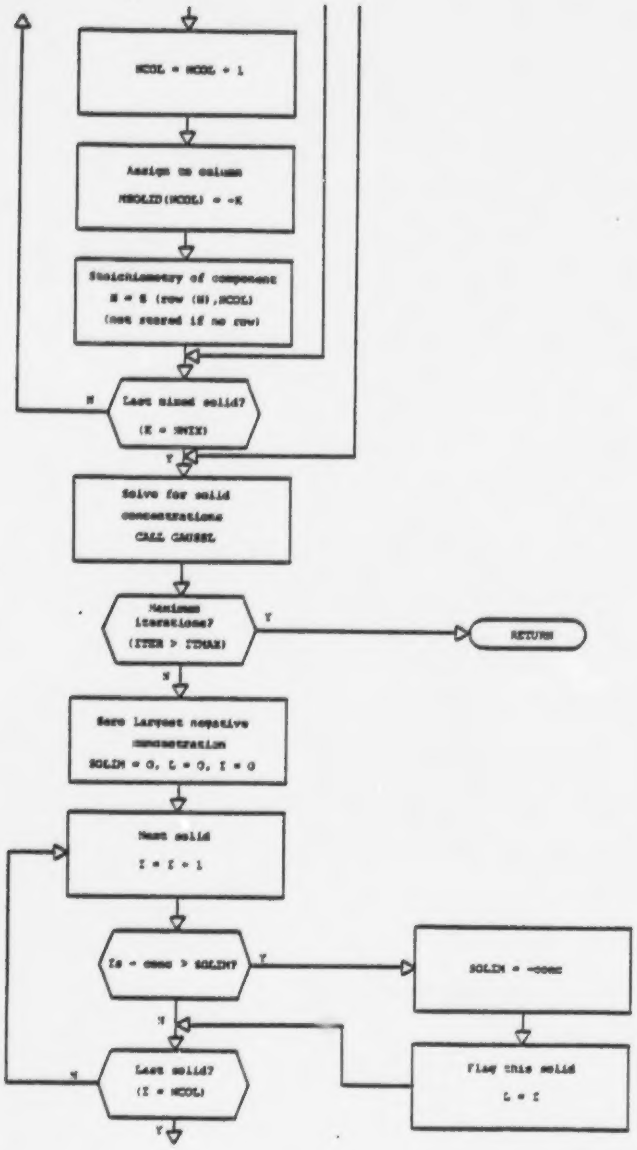
DISSOL is called by SUB3 to compute concentrations of imposed solids and mixed solids and to store them in the Y array. Additionally, if any are undersaturated, the most undersaturated is dissolved. Each of the N(=NSOL+NBMIX) imposed (precipitated) solids is assigned a value in the array MSOLID(K), K=1,N with MSOLID being a solid number if positive and a mixed solid number if negative.

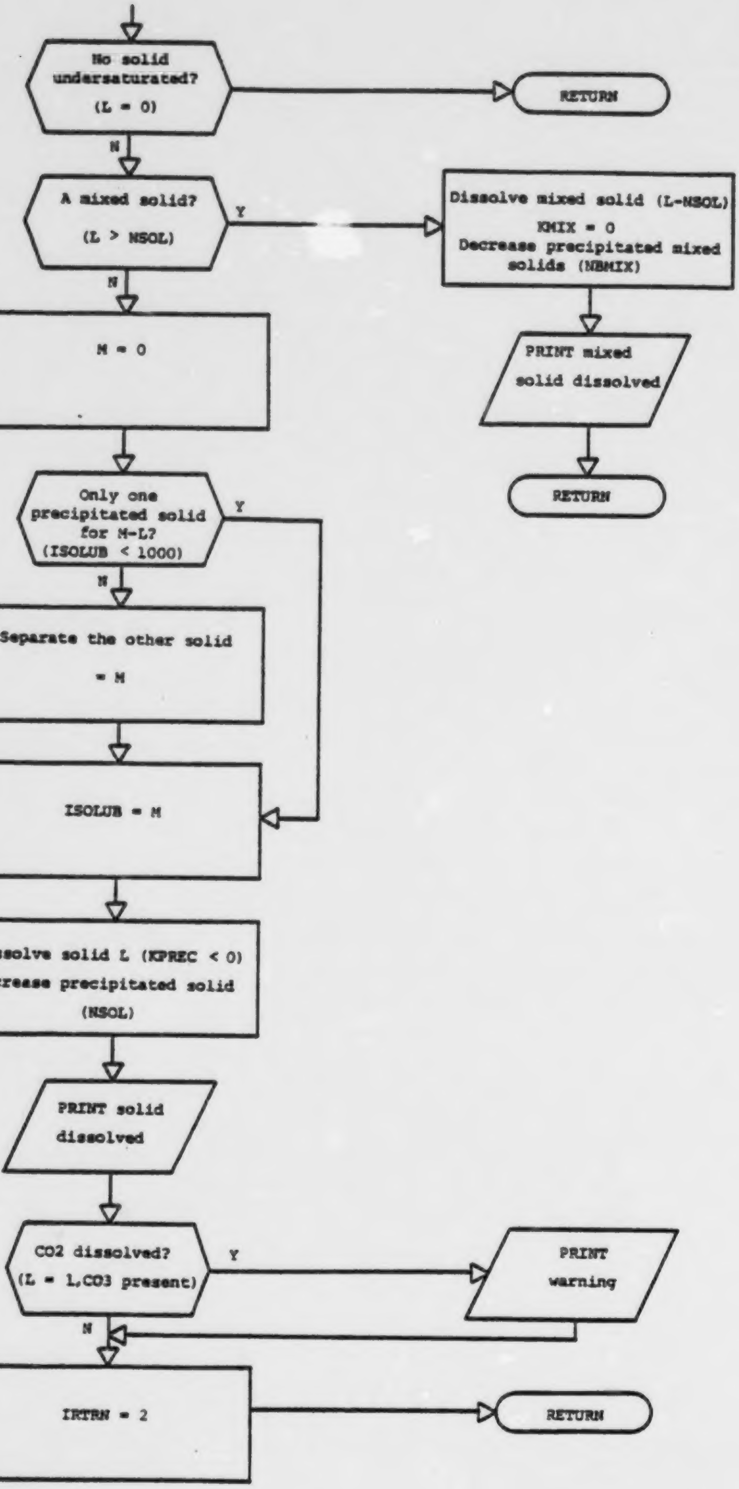

DISSOL
NCOL = number secondary variables (N2Y), IRTRN = 1, NO = precipitated solids + mixed precipitated solids
No precipitated solids or mixed solids? (NO =0)
Y
RETURN
N
Zero S matrix
For all variables, KMF = 1, NELEK = 0, JDUM = 0.
Not secondary variable flag, ISEC = 0
Flag secondary variables
(COEX < 0.001) ISEC = 1
ion-ion redox?
(MMLL > 0)
N
Y
K = 1
Next ion-ion redox
K = K + 1
Solids of JM will be solids of IM or E
KMF = KMR/KLR
NELEK = NELEC/KLR


Unflag one variable
If not JM,IM
If not IM,NSEC
ISEC = 0,ISEC(JM) = -IM
Transfer JM error (Y)
to IM and K
N
Last ion-ion Redox?
(K = NELL)
Y
NROW = 0
I = 1
I = I + 1
Secondary variable?
(ISEC > 0)
N
Y
NROW = NROW + 1
Assign Row to I (JDUM)
Amount of I in solid
= Y for row


I = NLAST?
N
Y
I = 1
I = I + 1
Second ion in ion-ion Redox? (ISEC < 0)
Y
Row = row (-ISEC)
N
I = NLAST?
N
Y
Zero MSOLID
NCOL = 0
No precipitated solids? (MSOL = 0)
Y
N
K = 0

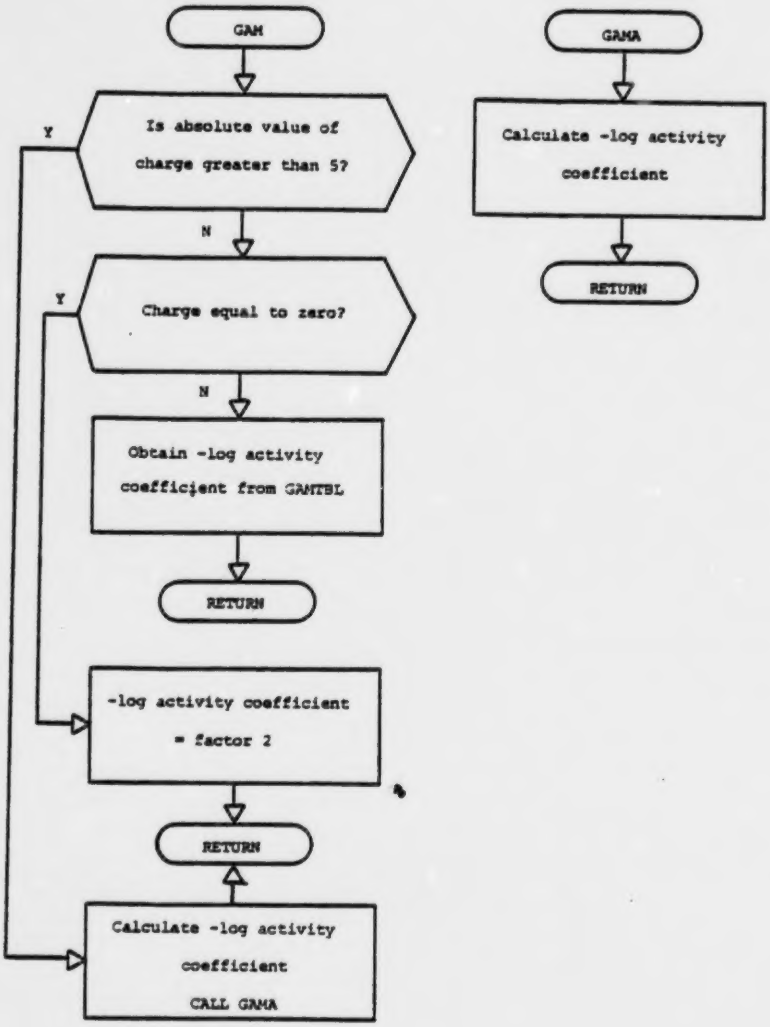

Next solid
K = K + 1
A precipitated solid?
(KPREC > 0)
N
Y
NCOL = NCOL + 1
Assign to column
NSOLID(NCOL) = K
Stoichiometry of component
I = S (row (I),NCOL)
(not stored if no row)
N
Last solid?
(K = NS)
Y
Any precipitated mixed solids?
(NMMIX > 0)
N
Y
K = 0
Next mixed solid
K = K + 1
A precipitated mixed solid?
(KMIX = 1)
N
Y


NCOL = NCOL + 1
Assign to column
MSOLID(NCOL) = -K
Stoichiometry of component
S = S (row (N),NCOL)
(not stored if no row)
N
Last mixed solid?
(K = NMIX)
Y
Solve for solid
concentrations
CALL GAUSSL
Maximum
iterations?
(ITER > ITMAX)
Y
RETURN
N
Zero largest negative
concentration
SOLIM = 0, L = 0, I = 0
Next solid
I = I + 1
Is - conc > SOLIM?
Y
SOLIM = -conc
N
Flag this solid
L = I
N
Last solid?
(I = NCOL)
Y


No solid undersaturated? (L = 0)
RETURN
N
A mixed solid? (L > NSOL)
Y
Dissolve mixed solid (L-NSOL)
KMIX = 0
Decrease precipitated mixed solids (NBMIX)
PRINT mixed solid dissolved
RETURN
N
M = 0
Only one precipitated solid for M-L? (ISOLUB < 1000)
Y
N
Separate the other solid = M
ISOLUB = M
Dissolve solid L (KPREC < 0)
Decrease precipitated solid (NSOL)
PRINT solid dissolved
CO2 dissolved? (L = L,CO3 present)
Y
PRINT warning
N
IRTRN = 2
RETURN

GAM and GAMA

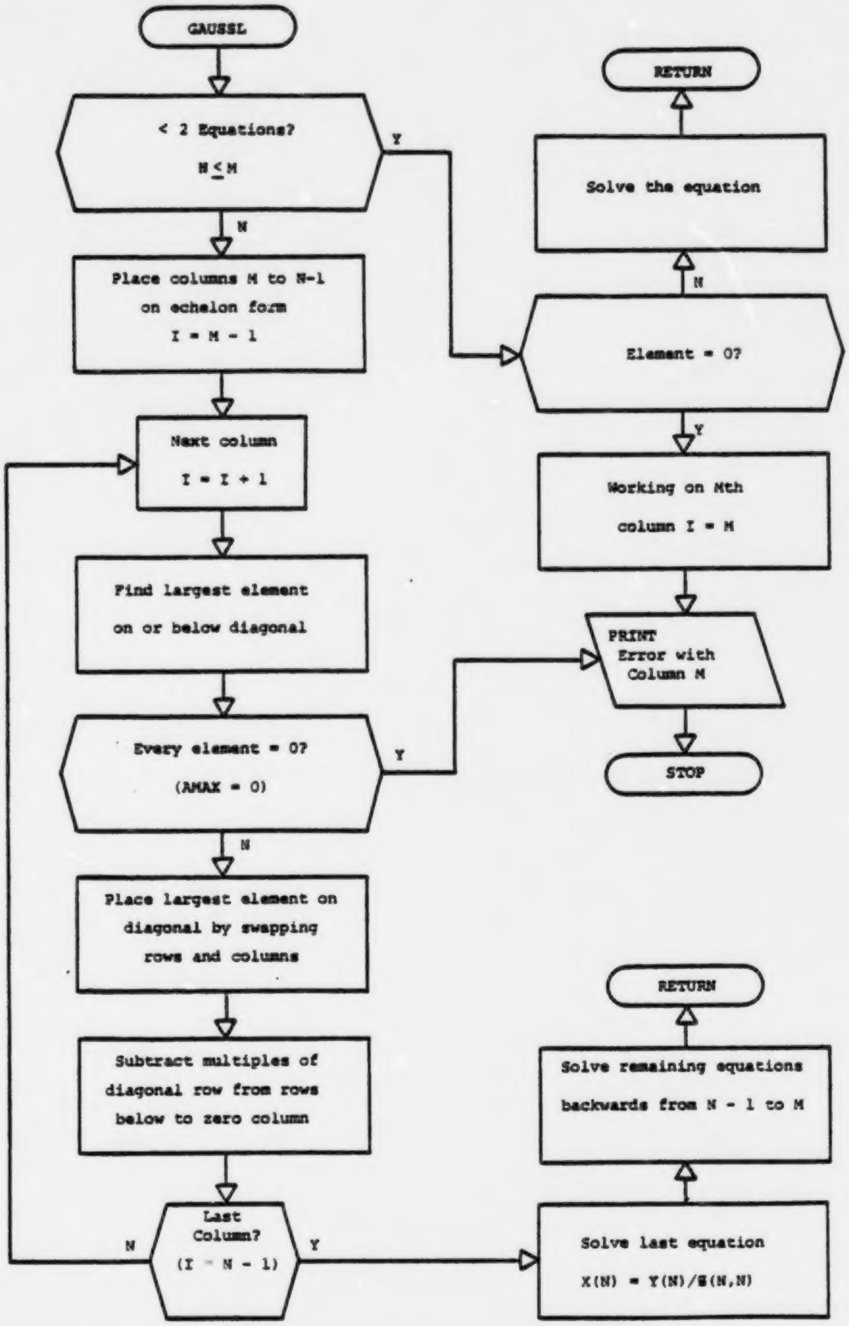
Functions GAM and GAMA are used by subroutine IONCOR to perform activity corrections to stability constants resulting from changes in the ionic strength of the solution. GAMA, called with charge M, is used each time IONCOR is entered to initialize a lookup table of correction factors (GAMTBL) for solids/complexes with absolute charges of 1 through 5, and by GAM when absolute charges greater than five are encountered, it uses factors from the Davies or Magnuson equations computed in IONCOR. Function GAM, called with charge M, is used within IONCOR to perform the corrections for individual solids and complexes.


GAM
Is absolute value of charge greater than 5?
Y
N
Charge equal to zero?
Y
N
Obtain -log activity coefficient from GAMTBL
RETURN
-log activity coefficient = factor 2
RETURN
Calculate -log activity coefficient CALL GAMA
GAMA
Calculate -log activity coefficient
RETURN

GAUSSL

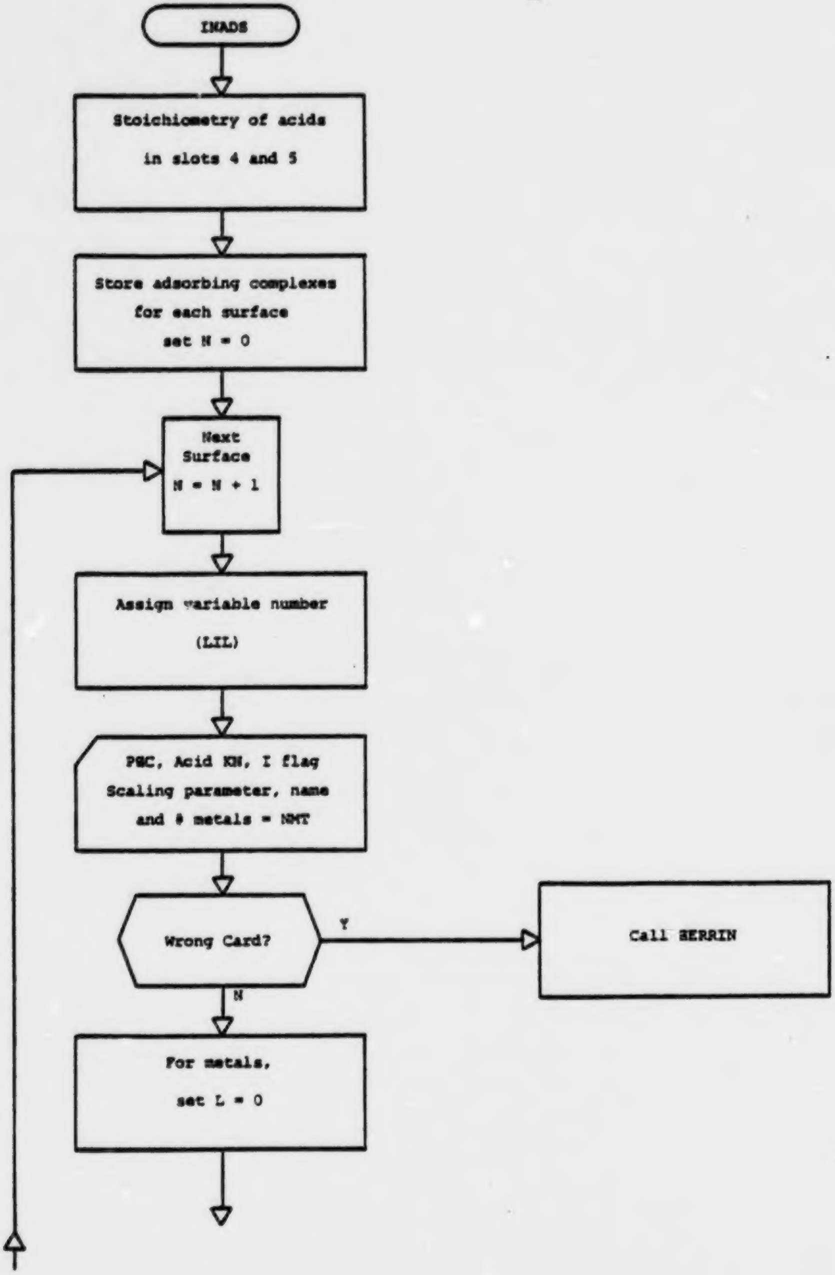
GAUSSL is called by DISSOL and SUB3 to solve a system of equations. When called by SUB3 it computes the adjustments to free concentrations from Z, the dependence of total concentration on free concentrations, and Y, the error in total concentrations. It solves equations M to N where M is 1 if pH calculated and 2 if pH fixed and N is NES if pE fixed or NEL if pE calculated. Equations for secondary variables are trivial (X=Y). The solutions are stored in X which is equivalenced to Y.

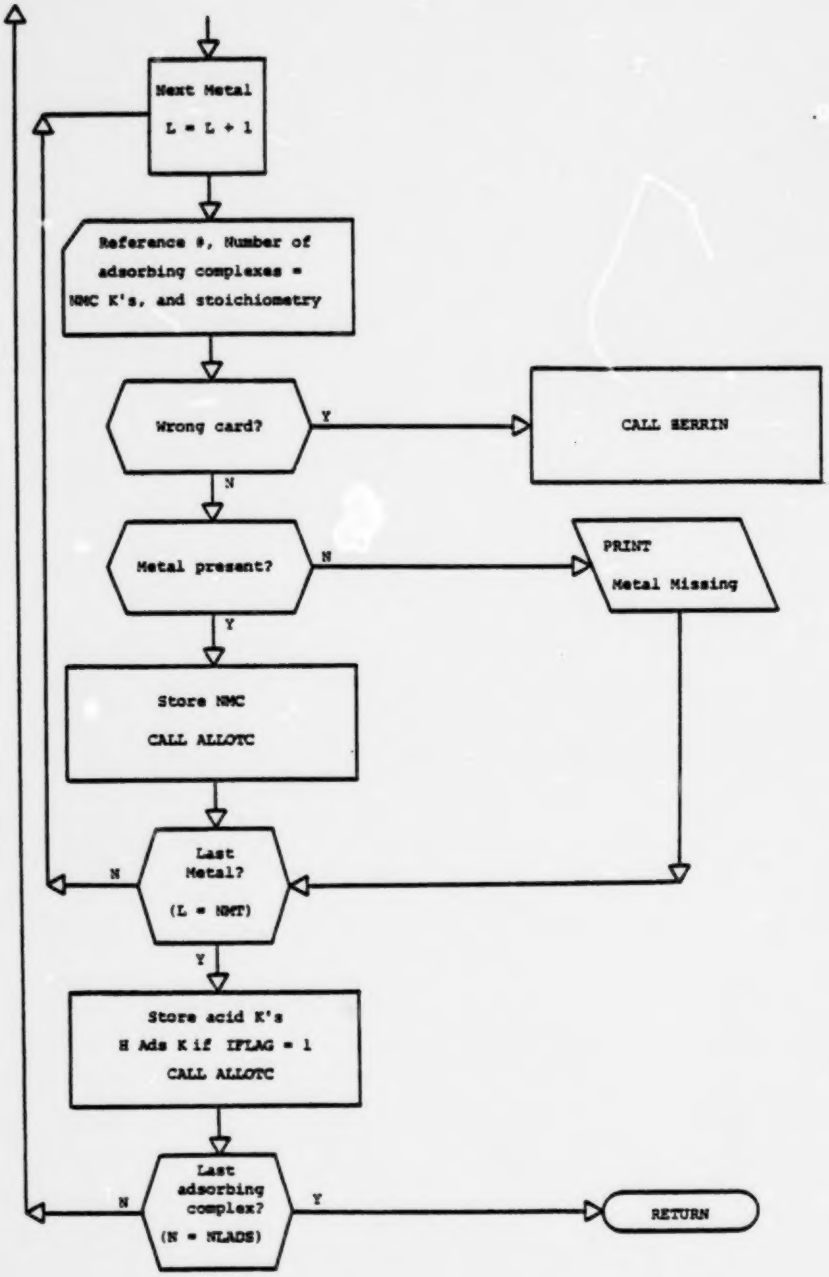
When called by DISSOL there are NSOL + NBMIX equations and GAUSSL computes the concentrations of solids from the stoichiometry of variables in solids (Z) and the amounts of secondary variables in solids (Y).


GAUSSL
< 2 Equations?
N ≤ M
Y
N
Place columns M to N-1
on echelon form
I = M - 1
Next column
I = I + 1
Find largest element
on or below diagonal
Every element = 0?
(AMAX = 0)
Y
N
Place largest element on
diagonal by swapping
rows and columns
Subtract multiples of
diagonal row from rows
below to zero column
Last
Column?
(I = N - 1)
N
Y
RETURN
Solve the equation
N
Element = 0?
Y
Working on Mth
column I = M
PRINT
Error with
Column M
STOP
RETURN
Solve remaining equations
backwards from N - 1 to M
Solve last equation
X(N) = Y(N)/S(N,N)

INADS

INADS is used by SUB1 to read adsorption data from the user deck when adsorption is being considered. The adsorbing surfaces are treated as ligands by the program.


INADS
Stoichiometry of acids in slots 4 and 5
Store adsorbing complexes for each surface set N = 0
Next Surface N = N + 1
Assign variable number (LIL)
PSC, Acid KN, I flag Scaling parameter, name and # metals = NMT
Wrong Card?
Y
Call HERRIN
N
For metals, set L = 0


Next Metal
L = L + 1
Reference #, Number of adsorbing complexes = NMC K's, and stoichiometry
Wrong card?
Y
CALL SERRIN
N
Metal present?
N
PRINT
Metal Missing
Y
Store NMC
CALL ALLOTC
Last Metal?
(L = NMT)
N
Y
Store acid K's
H Ads K if IFLAG = 1
CALL ALLOTC
Last adsorbing complex?
(N = NLADS)
N
Y
RETURN

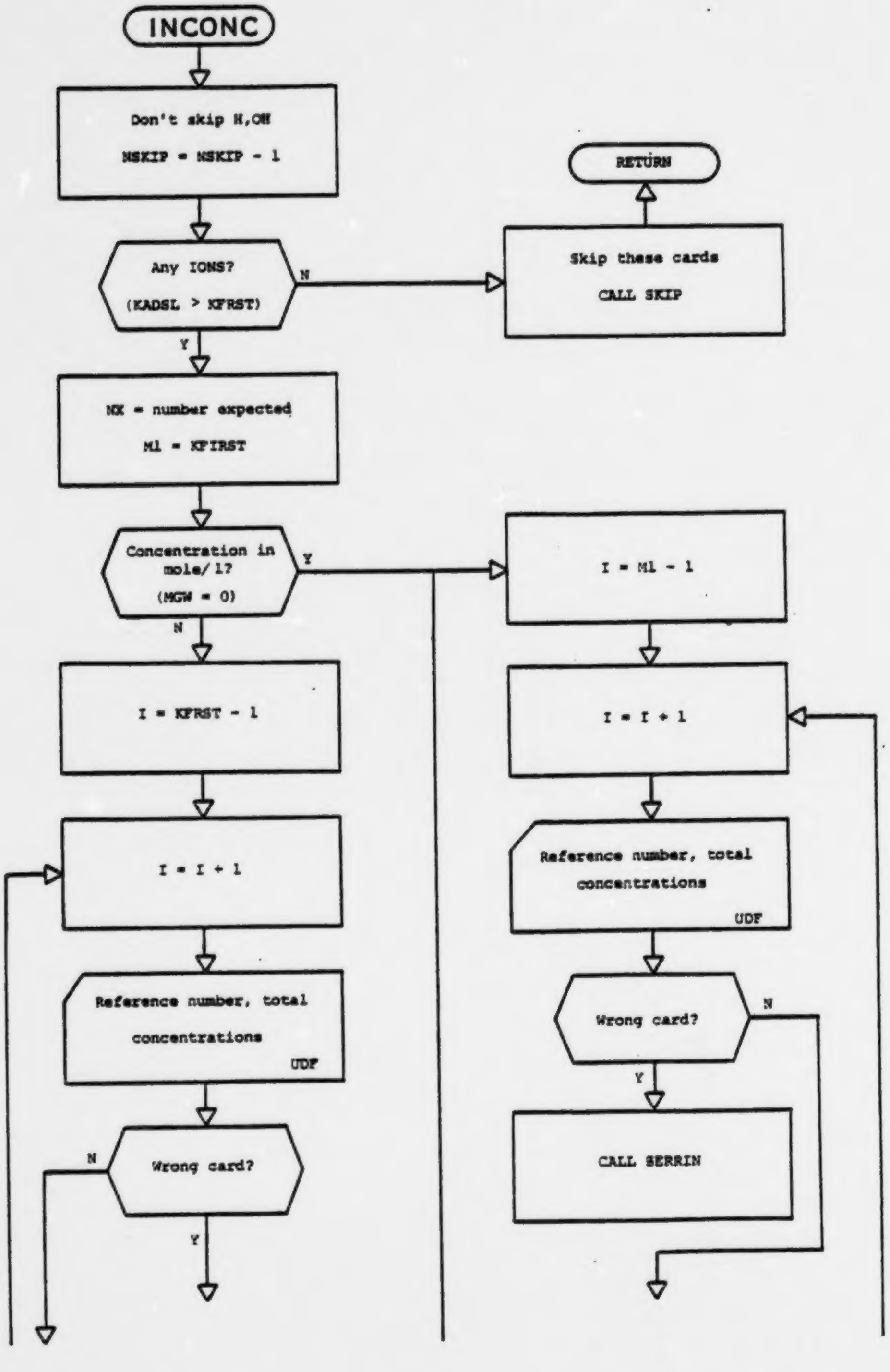
INCONC

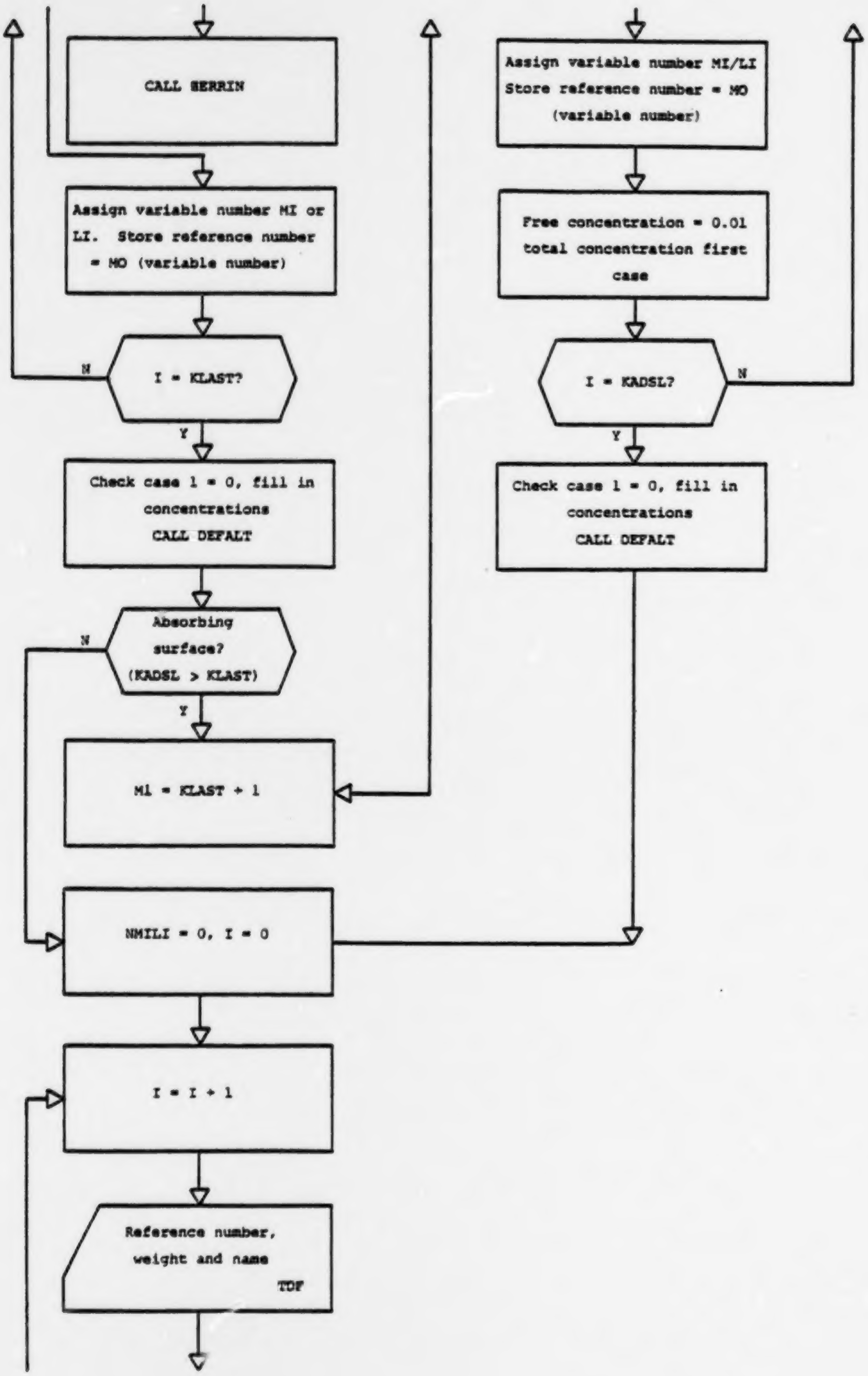
Subroutine INCONC is called by INMAT to read either metal or ligand concentration cards from the user's input deck. INCONC also reads metal and ligand charges, names, and molecular weights from the Thermodynamic Data File and converts concentrations from milligrams per liter to -log molar units if concentrations are being read in this form. INCONC is called with five parameters. The first two of these (KFRST and KLAST) describe the positions of the first and last ions among the variables. The third parameter (KADSL) gives the position of the last adsorbing surface; if no adsorbing surfaces are being considered, KADSL is equal to KLAST. The fourth parameter (NSKIP) is equal to NCM if metal cards and data base data are being read or to NCL if ligands are being processed. The fifth parameter (MILI) is MI when metals are being processed and LI when ligands are being processed. Each reference number is assigned to a variable number.

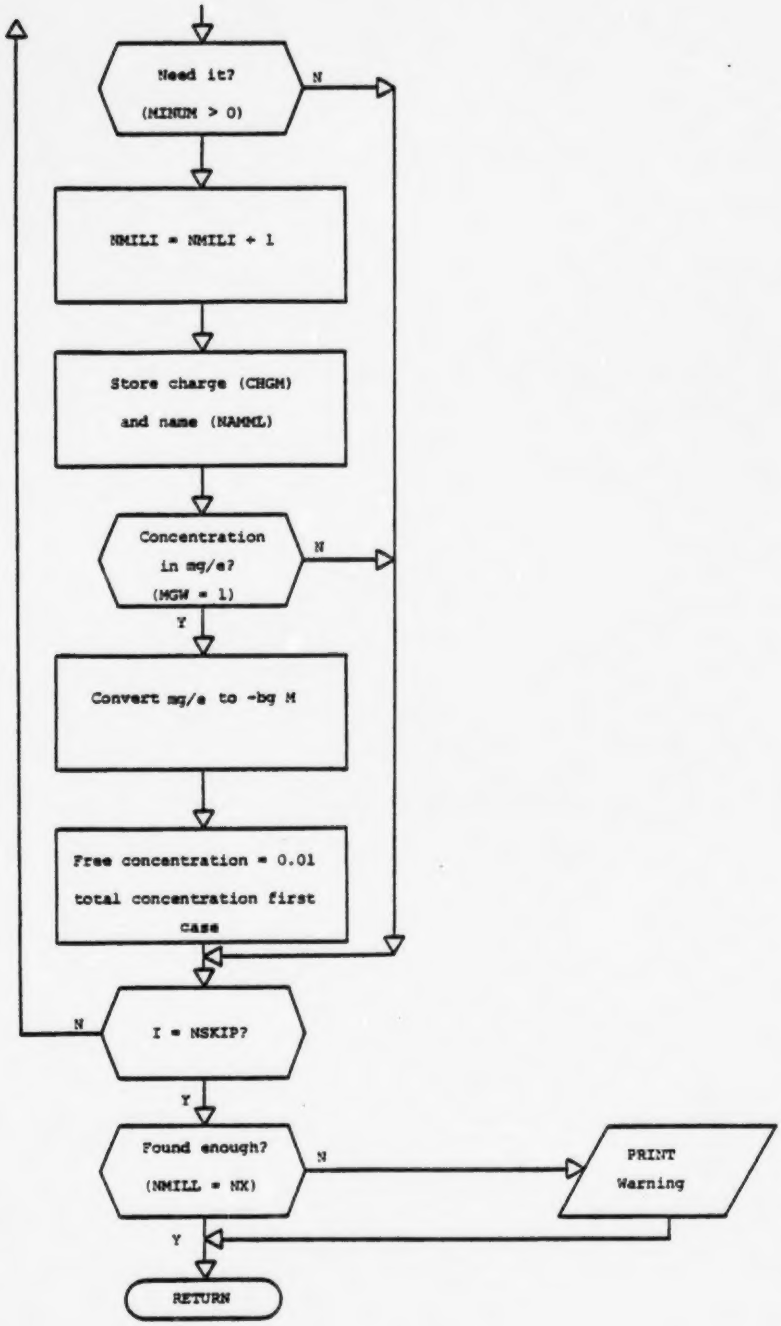
If KFRST = KLAST, there are no cards to be read, so the call to SKIP skips all but one line of this section of the Thermodynamic Data File; the line not skipped is either H^+ or OH^-, both of which are read in INMAT.

If data are being read in milligram per liter form, the concentrations are stored in array TMINW. In the case of -log molar concentrations, input data are read into TMIN.

The appropriate section of the Thermodynamic Data File is read to obtain a charge, molecular weight, and a name for each ion in the system. These are stored in the arrays CHGM, WGM, and NAMML respectively. If input was given in mg/liter form, the -log molar concentration is computed and stored in array TMIN.


INCONC
Don't skip H,OH
NSKIP = NSKIP - 1
Any IONS?
(KADSL > KFRST)
N
Y
Skip these cards
CALL SKIP
RETURN
MX = number expected
M1 = KFIRST
Concentration in mole/l?
(MGW = 0)
Y
N
I = KFRST - 1
I = I + 1
Reference number, total concentrations
UDF
Wrong card?
N
Y
I = M1 - 1
I = I + 1
Reference number, total concentrations
UDF
Wrong card?
N
Y
CALL SERRIN


CALL SERRIN
Assign variable number MI or LI. Store reference number = MO (variable number)
N
I = KLAST?
Y
Check case 1 = 0, fill in concentrations
CALL DEFALT
N
Absorbing surface? (KADSL > KLAST)
Y
M1 = KLAST + 1
NMILI = 0, I = 0
I = I + 1
Reference number, weight and name
TDF
Assign variable number MI/LI
Store reference number = MO (variable number)
Free concentration = 0.01 total concentration first case
I = KADSL?
N
Y
Check case 1 = 0, fill in concentrations
CALL DEFALT


Need it?
(MINUM > 0)
N
NMILI = NMILI + 1
Store charge (CHGM)
and name (NAMML)
Concentration
in mg/e?
(MGW = 1)
N
Y
Convert mg/e to -bg M
Free concentration = 0.01
total concentration first
case
N
I = NSKIP?
Y
Found enough?
(NMILL = NX)
N
PRINT
Warning
Y
RETURN

INMAT

INMAT is used by SUB1 to perform input of general parameters, a host of flags and counters, etc., as well as input of metals and ligand data from both the user card deck and the Thermodynamic Data File. The first card read from the user deck contains the following flags and counters (in the order they appear on the card):

ICHAR	A character used for error detection (1).
NM	Number of metals to be in the system (including H^+).
NL3	Number of ligands (including OH^-).
NCASES	Number of cases to be considered.
NPH	pH flag (NPH=0: pH fixed; NPH=1: pH computed).
ION	Ionic strength flag (ION=0: ionic strength is fixed; ION=1: ionic strength is computed).
JFIX	Charge balance flag (JFIX=100: concentrations of all ligands may be varied by the program in order to achieve electroneutrality; JFIX=0: no attempt is to be made by the program to achieve electroneutrality; 99>JFIX>0: the ion number "JFIX" may be varied to achieve charge balance).
ICHG	Charge on hydrolyzed ion JFIX (negative for ligand, positive for metal).
NALK	Carbon flag (NALK=0: carbonate input; NALK=1: alkalinity input; NALK=2: PCO2 to be given with large carbonate).
KREDOX	Redox flag (KREDOX=0: no redox considered; KREDOX=1: redox considered and redox potential (pE) is fixed; KREDOX=2: redox considered and pE is variable).
NIDS	Number of solids to be imposed, -1 means none allowed to precipitate.

NBMIX	Mixed solid flag (NBMIX=0: no mixed solids can precipitate; NBMIX: all mixed solids allowed to precipitate; NBMIX=2: all allowed, except some to be specified as imposed or disallowed; NBMIX=3: none allowed, but saturation ratios calculated).
MGW	Milligram per liter or -log M input flag (MGW=0: -log M concentrations used in input; MGW=1: milligrams/liter used).
NLADS	Number of absorbing surfaces in the system.
IJAC	Interaction intensity/capacity flag (IJAC=0: neither will be reported; IJAC=1: capacities reported; IJAC=2: intensities reported; IJAC=3: both reported).
IGAM	Activities calculation flag (IGAM=0: Magnuson; IGAM=1: Davies)
NRS	Saturation ratio output selection (NRS=1: no saturation ratios)
IACT	Activities/concentrations flag (IACT=0: complex concentrations reported as -log M; IACT=1: concentrations reported both as -log M and as -log activity).
117	Complex concentration output selection (117-1: routine OUT17 not used).
ICON	Thermodynamic data output flag (ICON=2: no data reported; ICON=1: only uncorrected data reported; ICON=0: both corrected and uncorrected data reported).

Following this, a number of the input values are checked for validity and the following variables are set:

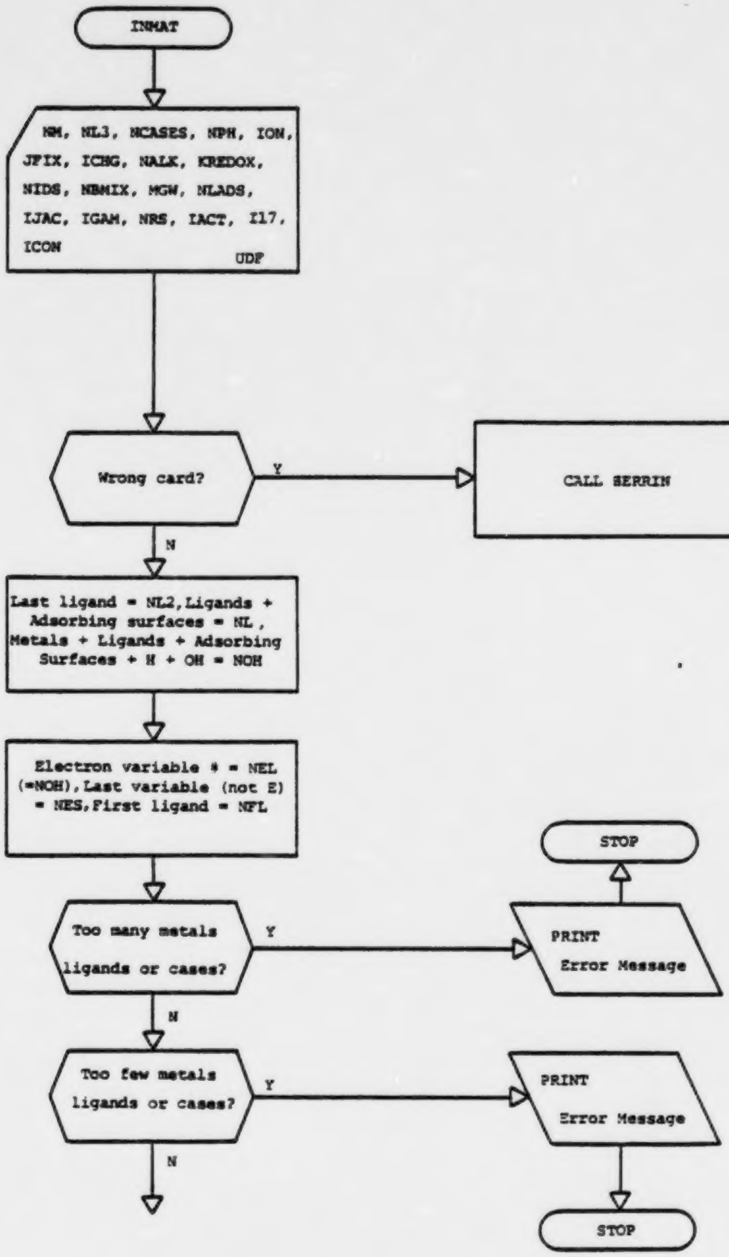
NIQ	Pointer to the last ion in the system (excluding OH^-).
NFRST	Pointer to the first ion whose free concentration will be computed.
NLAST	Pointer to the last ion whose free concentration will be computed.
NES	Pointer to last ligand or surface in the system.
NOH	Pointer to OH^- (=NM+NL3+NLADS).

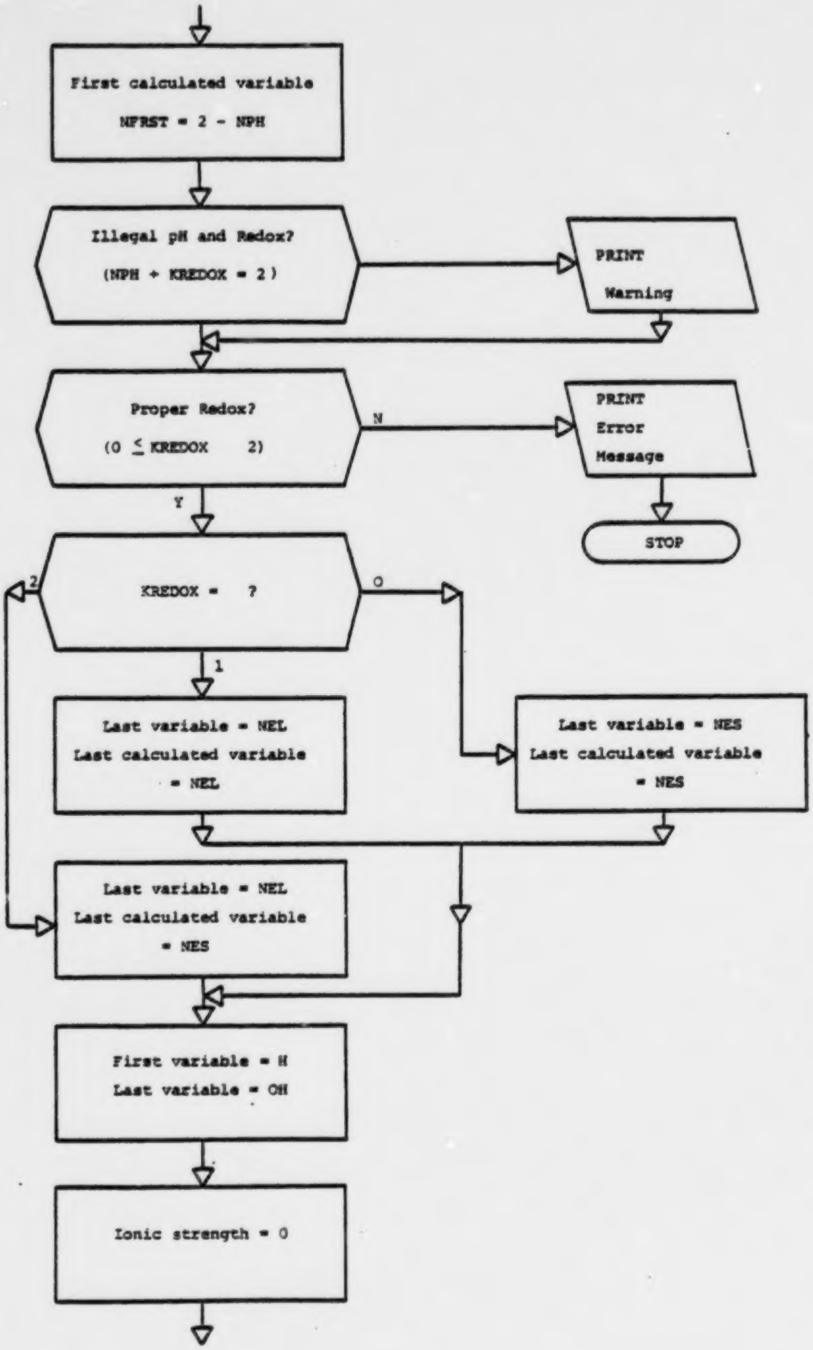
Hydrogen is then established as the first variable in the system (MO(1)=50, MI(50)=1), and hydroxide as the last variable (MO(NOH)=99, LI(99)=NOH). More data from the user data file are then read.

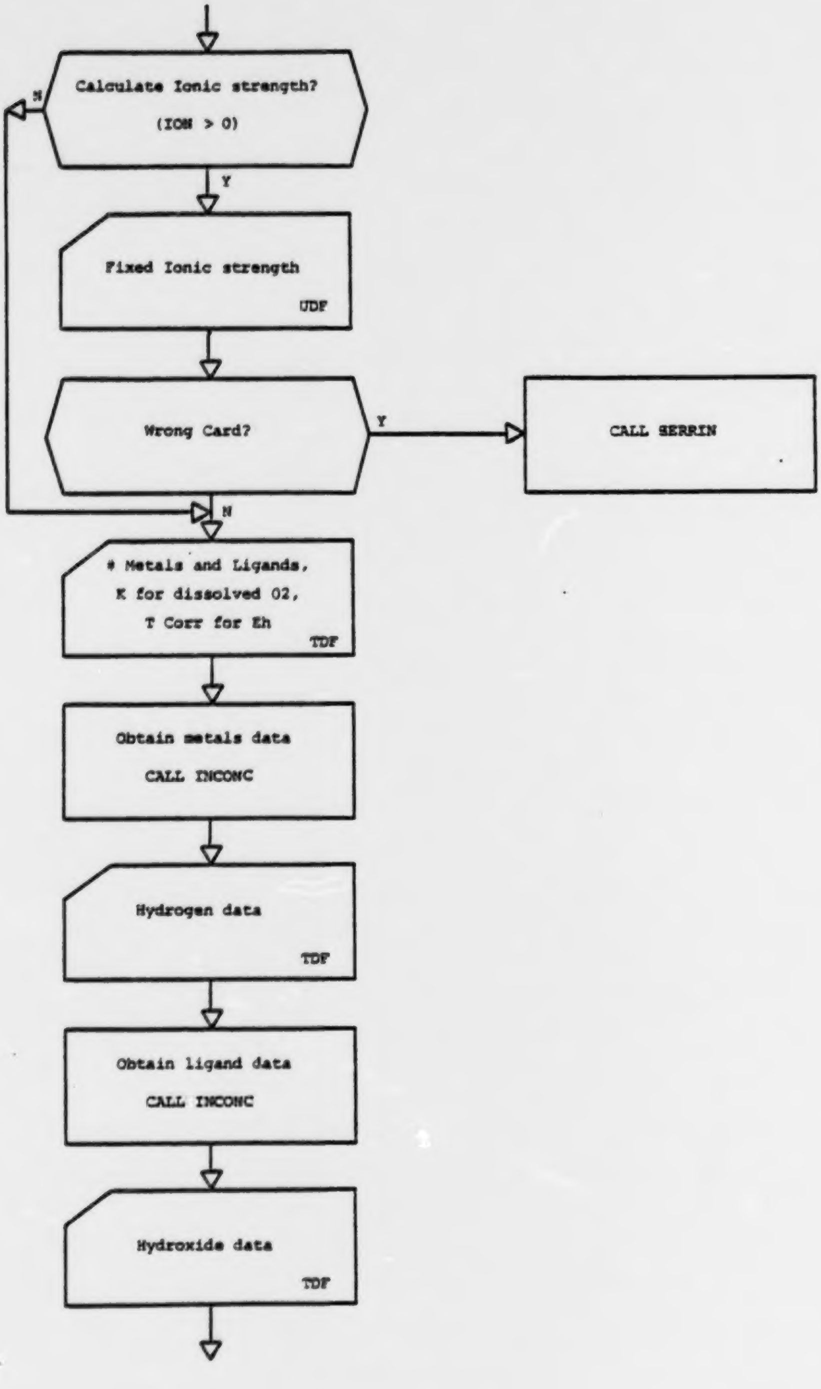
When INMAT is finished with the user data deck, the remainder of the routine is concerned with obtaining data for complexes and solids from the thermodynamic data base. There are a number of variables of import, among them are:

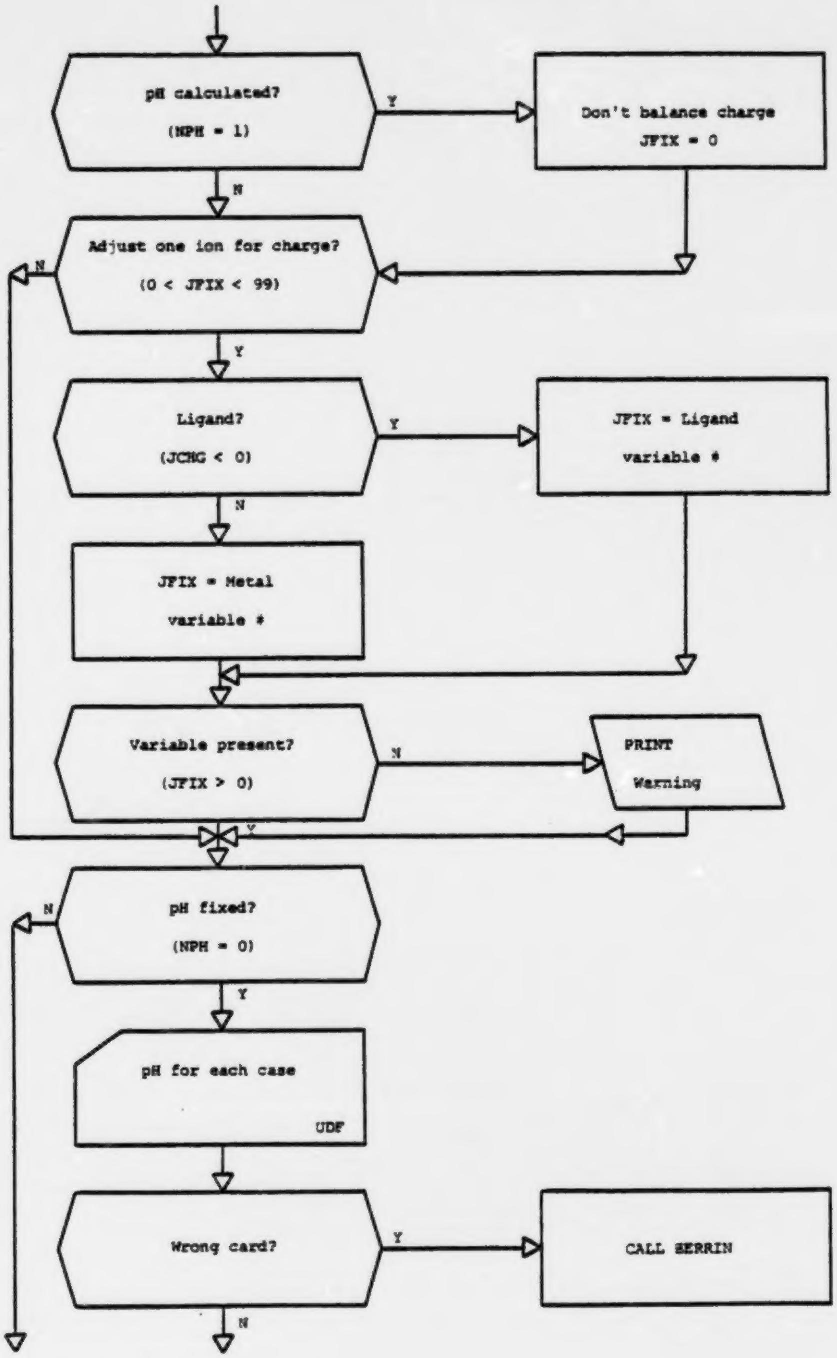
NMAT	One-half the number of lines of solid/complex data in the thermodynamic data base.
WK	-log of the association constant for water.
PCO2CT	-log of the reaction $2H^+ + CO_3^{2-} \longrightarrow CO_2(g)$
AMOD	Correction factor to be added to the constant "A" in the Davies or Magnuson equations if the thermodynamic data have been compiled for a temperature other than 25 degrees C.
NSOL	Precipitated solid count.
NS	Solid count.
NC	Complex count.
ND	Set to 1 if any are solids disallowed, otherwise zero.

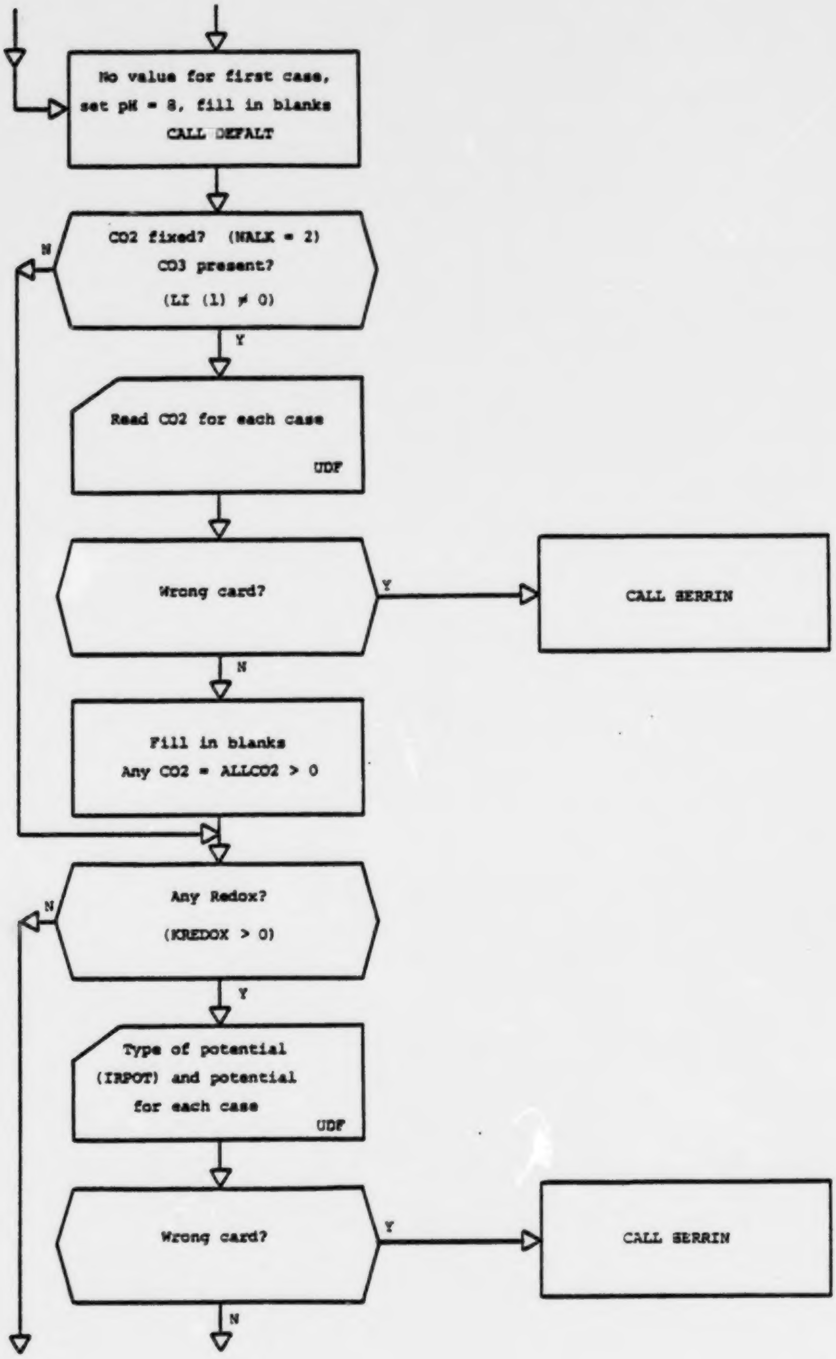
Data from the thermodynamic file which are appropriate to the system are stored by ALLOTC and ALLOTS. Solids have KPREC=-1 unless they can never precipitate (KPREC=0) or are already precipitated (KPREC=1). Metal-ligand pairs which have precipitated solids are flagged by setting ISOLUB (M,L) to the precipitated solid number N or (1000xN+N2) if two solids N and N2 have precipitated. The solid is also flagged with KPREC=1, and MSOLID(N) is the solid number of the Nth precipitated solid.

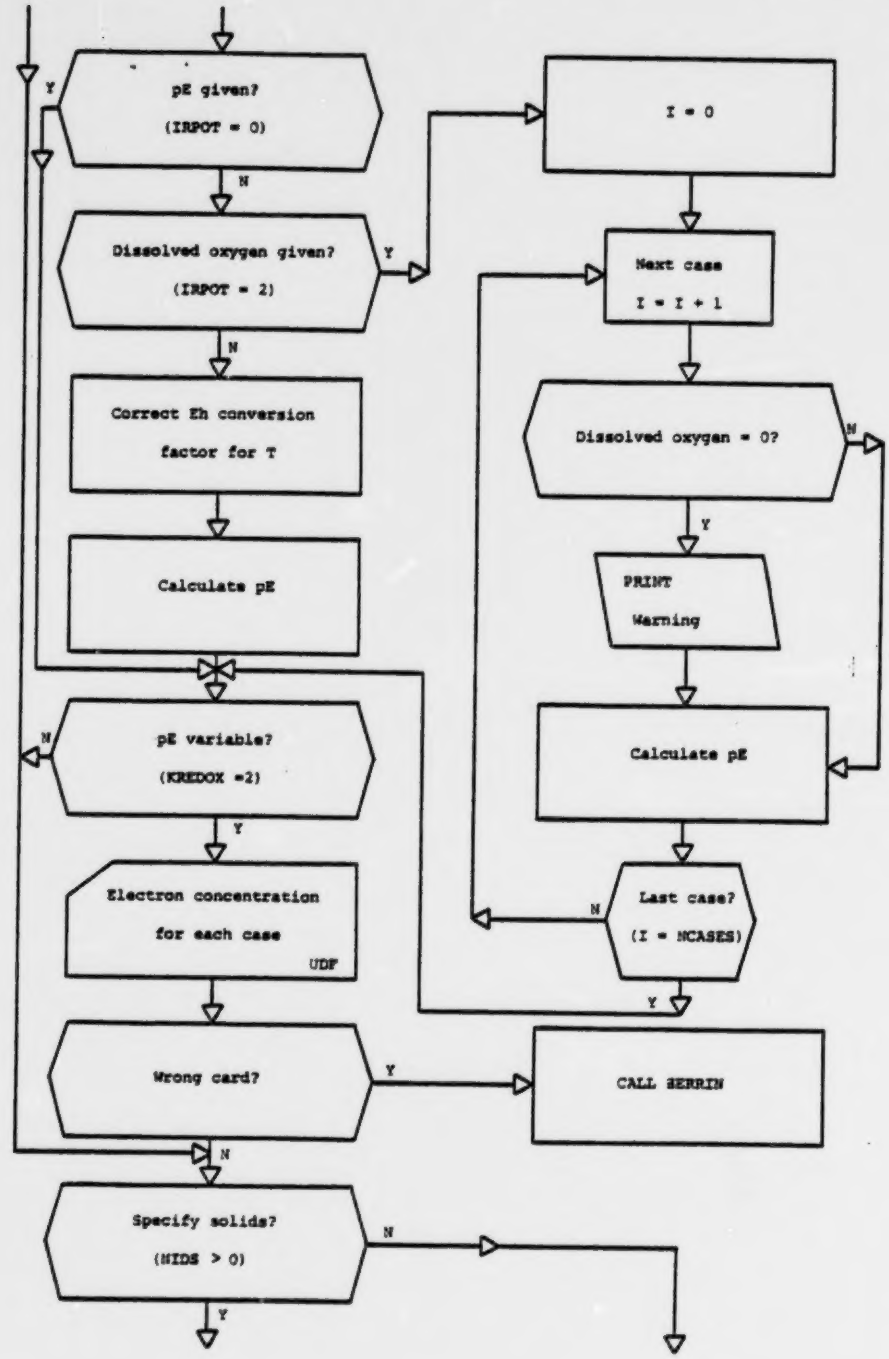

INMAT
NM, NL3, NCASES, NPH, ION,
JFIX, ICHG, NALK, KREDOX,
NIDS, NBMIX, MGW, NLADS,
IJAC, IGAM, NRS, IACT, I17,
ICON
UDF
Wrong card?
Y
CALL SERRIN
N
Last ligand = NL2, Ligands + Adsorbing surfaces = NL, Metals + Ligands + Adsorbing Surfaces + H + OH = NOH
Electron variable # = NEL (=NOH), Last variable (not E) = NES, First ligand = NFL
Too many metals ligands or cases?
Y
PRINT Error Message
STOP
N
Too few metals ligands or cases?
Y
PRINT Error Message
STOP
N

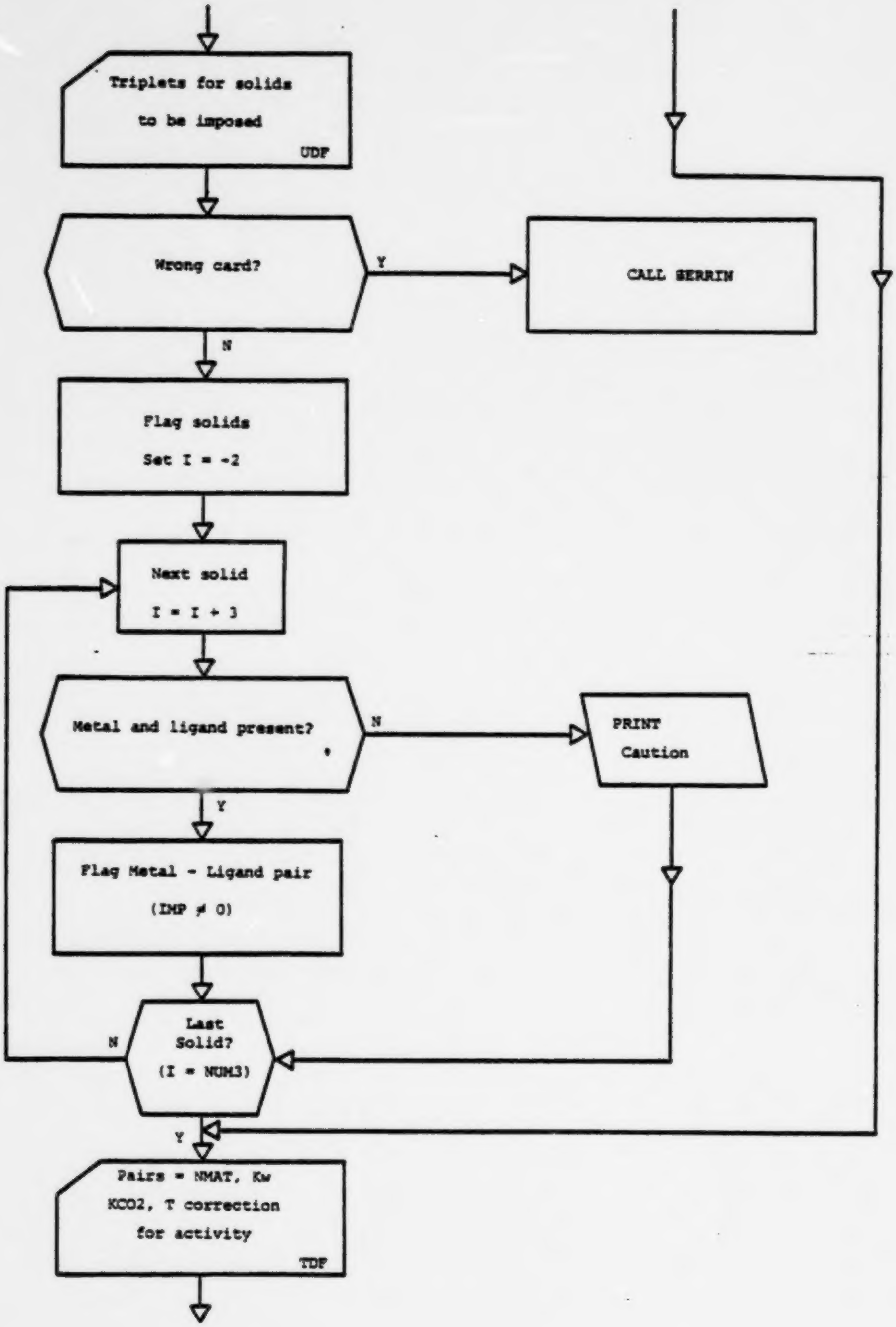

First calculated variable
NFRST = 2 - NPH
Illegal pH and Redox?
(NPH + KREDOX = 2)
PRINT
Warning
Proper Redox?
(0 ≤ KREDOX 2)
N
PRINT
Error
Message
STOP
Y
KREDOX = ?
2
0
1
Last variable = NEL
Last calculated variable
= NEL
Last variable = NES
Last calculated variable
= NES
Last variable = NEL
Last calculated variable
= NES
First variable = H
Last variable = OH
Ionic strength = 0

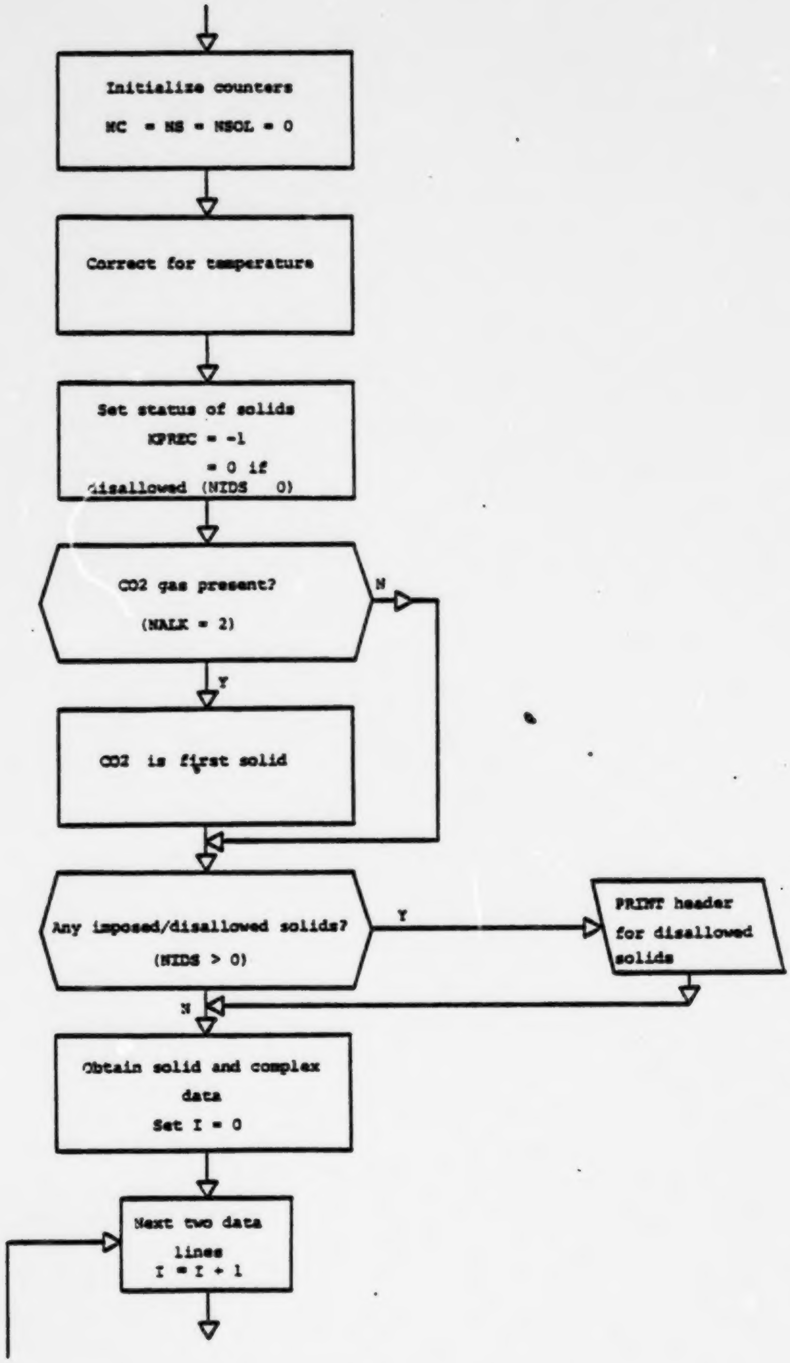

Calculate Ionic strength?
(ION > 0)
N
Y
Fixed Ionic strength
UDF
Wrong Card?
Y
CALL SERRIN
N
# Metals and Ligands,
K for dissolved O2,
T Corr for Eh
TDF
Obtain metals data
CALL INCONC
Hydrogen data
TDF
Obtain ligand data
CALL INCONC
Hydroxide data
TDF

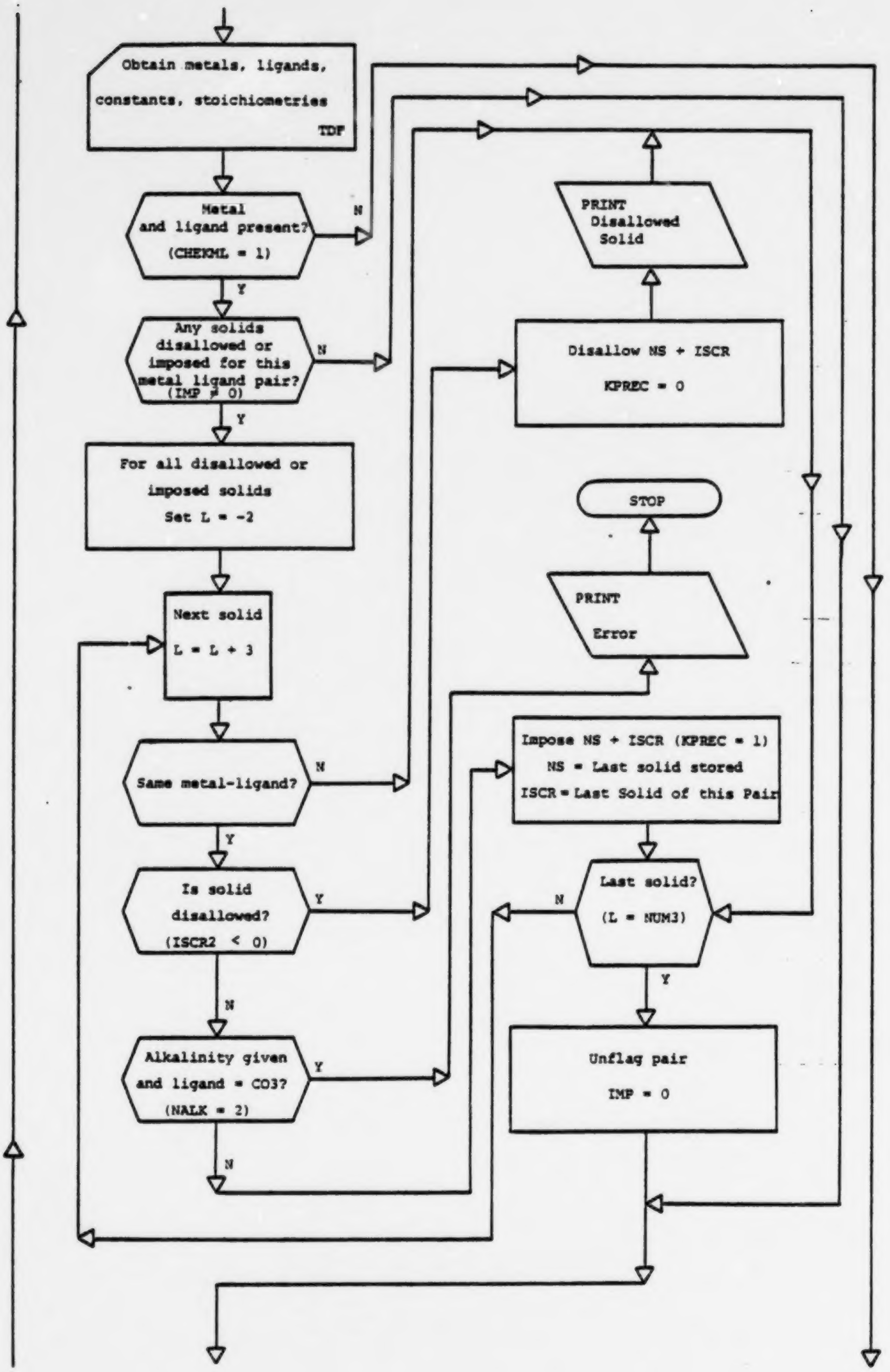

pH calculated?
(NPH = 1)
Y
Don't balance charge
JFIX = 0
N
Adjust one ion for charge?
(0 < JFIX < 99)
N
Y
Ligand?
(JCHG < 0)
Y
JFIX = Ligand
variable #
N
JFIX = Metal
variable #
Variable present?
(JFIX > 0)
N
PRINT
Warning
Y
pH fixed?
(NPH = 0)
N
Y
pH for each case
UDF
Wrong card?
Y
CALL SERRIN
N

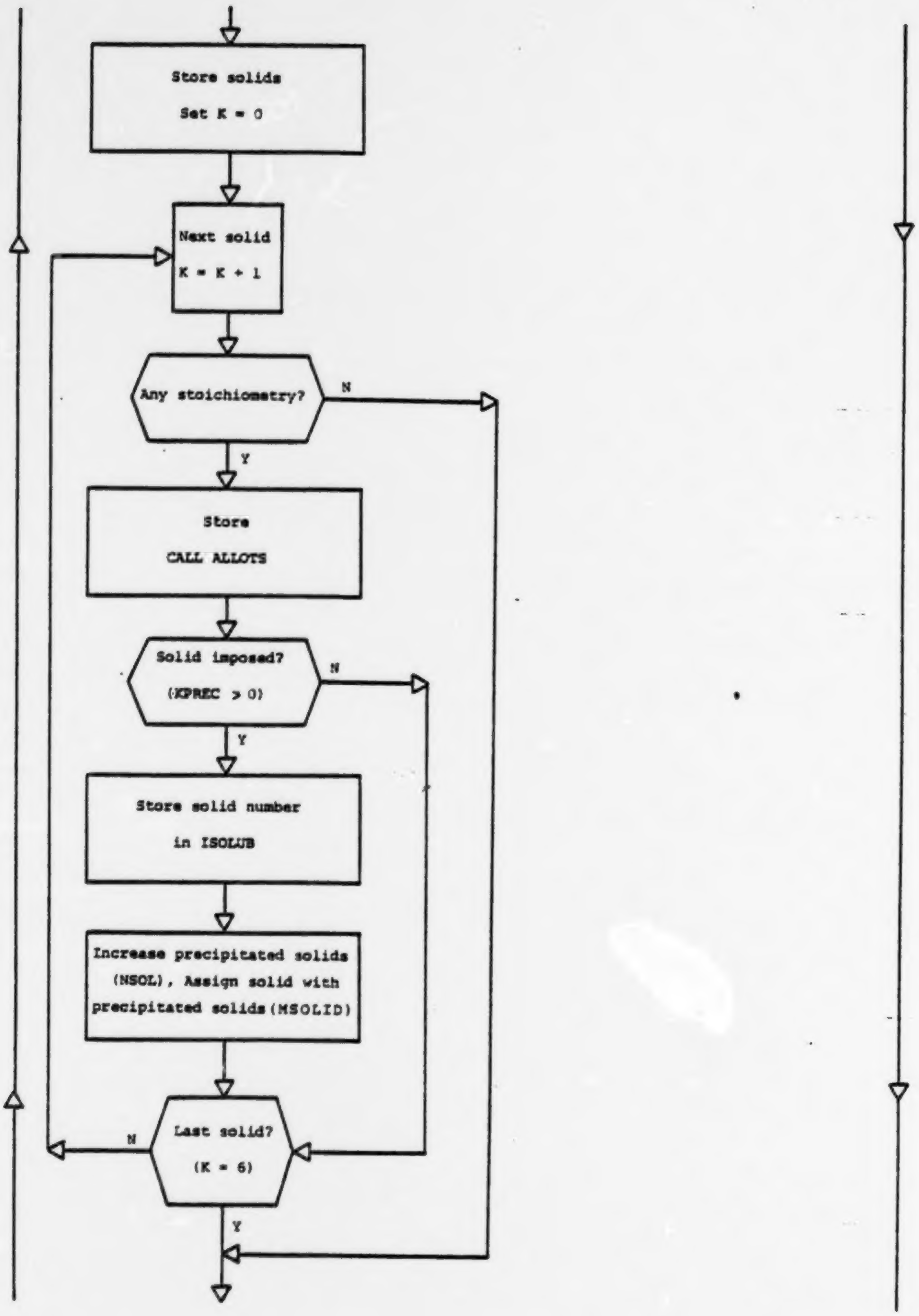

No value for first case,
set pH = 8, fill in blanks
CALL DEFALT
CO2 fixed? (NALK = 2)
CO3 present?
(LI (1) ≠ 0)
N
Y
Read CO2 for each case
UDF
Wrong card?
Y
CALL SERRIN
N
Fill in blanks
Any CO2 = ALLCO2 > 0
Any Redox?
(KREDOX > 0)
N
Y
Type of potential
(IRPOT) and potential
for each case
UDF
Wrong card?
Y
CALL SERRIN
N

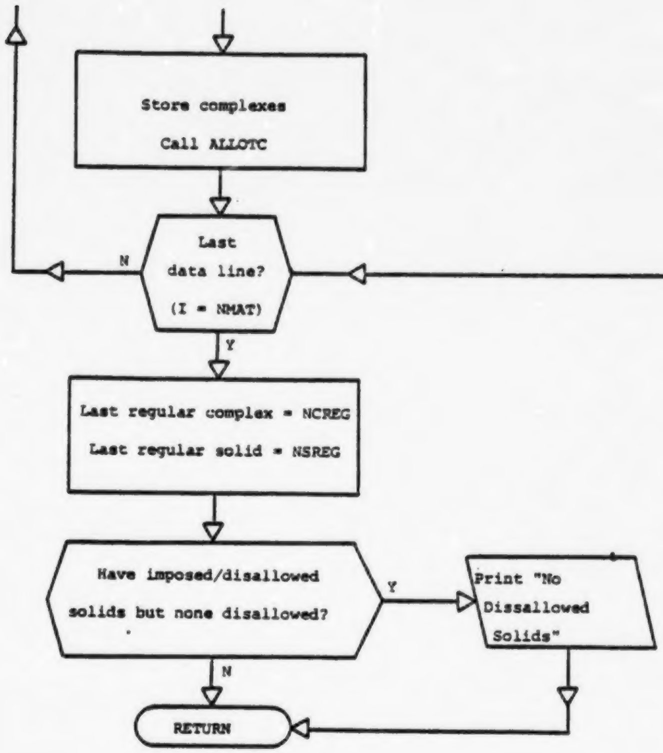

pE given?
(IRPOT = 0)
Y
N
Dissolved oxygen given?
(IRPOT = 2)
Y
N
Correct Eh conversion
factor for T
Calculate pE
I = 0
Next case
I = I + 1
Dissolved oxygen = 0?
N
Y
PRINT
Warning
Calculate pE
Last case?
(I = NCASES)
N
Y
pE variable?
(KREDOX =2)
N
Y
Electron concentration
for each case
UDF
Wrong card?
Y
CALL SERRIN
N
Specify solids?
(NIDS > 0)
N
Y


Triplets for solids
to be imposed
UDF
Wrong card?
Y
CALL SERRIN
N
Flag solids
Set I = -2
Next solid
I = I + 3
Metal and ligand present?
N
PRINT
Caution
Y
Flag Metal - Ligand pair
(IMP ≠ 0)
N
Last
Solid?
(I = NUM3)
Y
Pairs = NMAT, Kw
KCO2, T correction
for activity
TDF


Initialize counters
NC = NS = NSOL = 0
Correct for temperature
Set status of solids
KPREC = -1
= 0 if
disallowed (NIDS 0)
CO2 gas present?
(NALK = 2)
N
Y
CO2 is first solid
Any imposed/disallowed solids?
(NIDS > 0)
Y
PRINT header
for disallowed
solids
N
Obtain solid and complex
data
Set I = 0
Next two data
lines
I = I + 1


Obtain metals, ligands, constants, stoichiometries
TDF
Metal and ligand present? (CHEKML = 1)
N
Y
Any solids disallowed or imposed for this metal ligand pair? (IMP ≠ 0)
N
Y
For all disallowed or imposed solids Set L = -2
Next solid L = L + 3
Same metal-ligand?
N
Y
Is solid disallowed? (ISCR2 < 0)
Y
N
Alkalinity given and ligand = CO3? (NALK = 2)
Y
N
PRINT Disallowed Solid
Disallow NS + ISCR KPREC = 0
STOP
PRINT Error
Impose NS + ISCR (KPREC = 1) NS = Last solid stored ISCR = Last Solid of this Pair
Last solid? (L = NUM3)
N
Y
Unflag pair IMP = 0


Store solids
Set K = 0
Next solid
K = K + 1
Any stoichiometry?
N
Y
Store
CALL ALLOTS
Solid imposed?
(KPREC > 0)
N
Y
Store solid number
in ISOLUB
Increase precipitated solids
(NSOL), Assign solid with
precipitated solids(MSOLID)
Last solid?
(K = 6)
N
Y

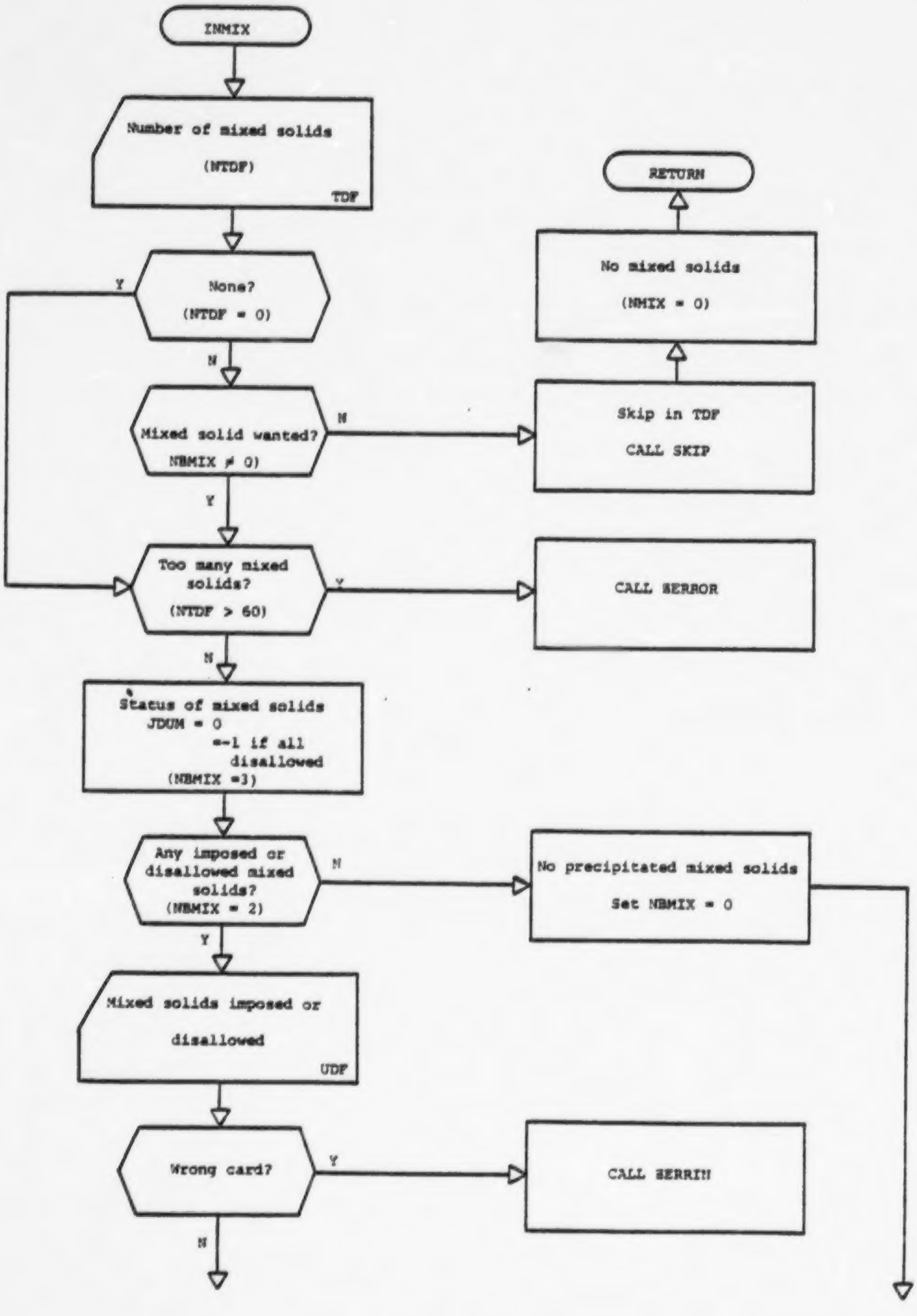

Store complexes
Call ALLOTC
Last data line? (I = NMAT)
N
Y
Last regular complex = NCREG
Last regular solid = NSREG
Have imposed/disallowed solids but none disallowed?
Y
N
Print "No Dissallowed Solids"
RETURN

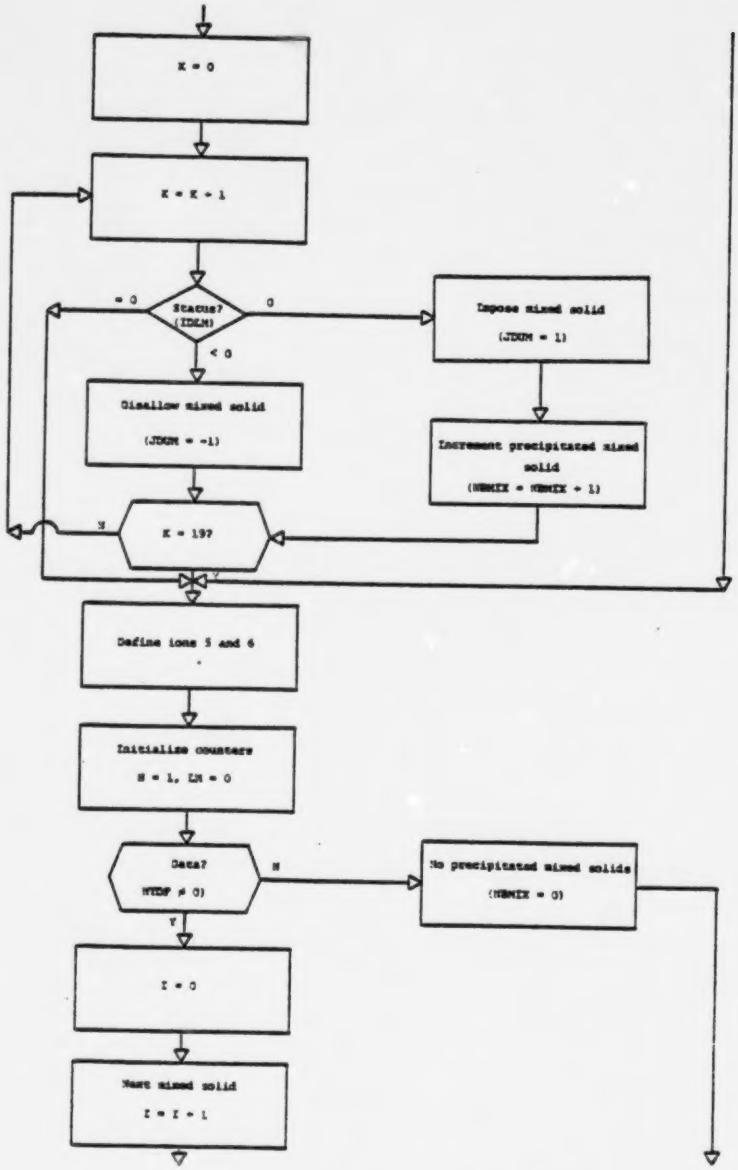
INMIX

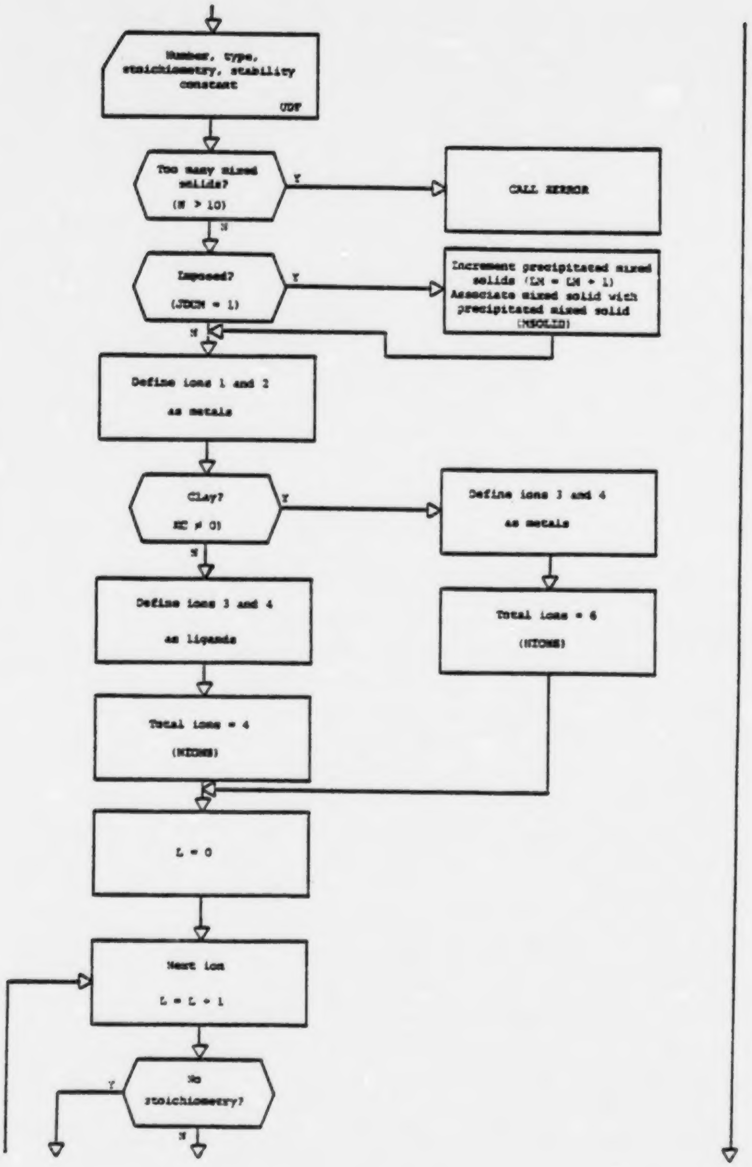
INMIX is called by SUB1 to process input of data for mixed solids. If a mixed solid is disallowed, JDUM(K) is set to -1 (where K is the reaction number of the mixed solid); if the mixed solid is imposed, JDUM(K) is set to 1. When data are found for a mixed solid which is in the system, they are stored as follows:

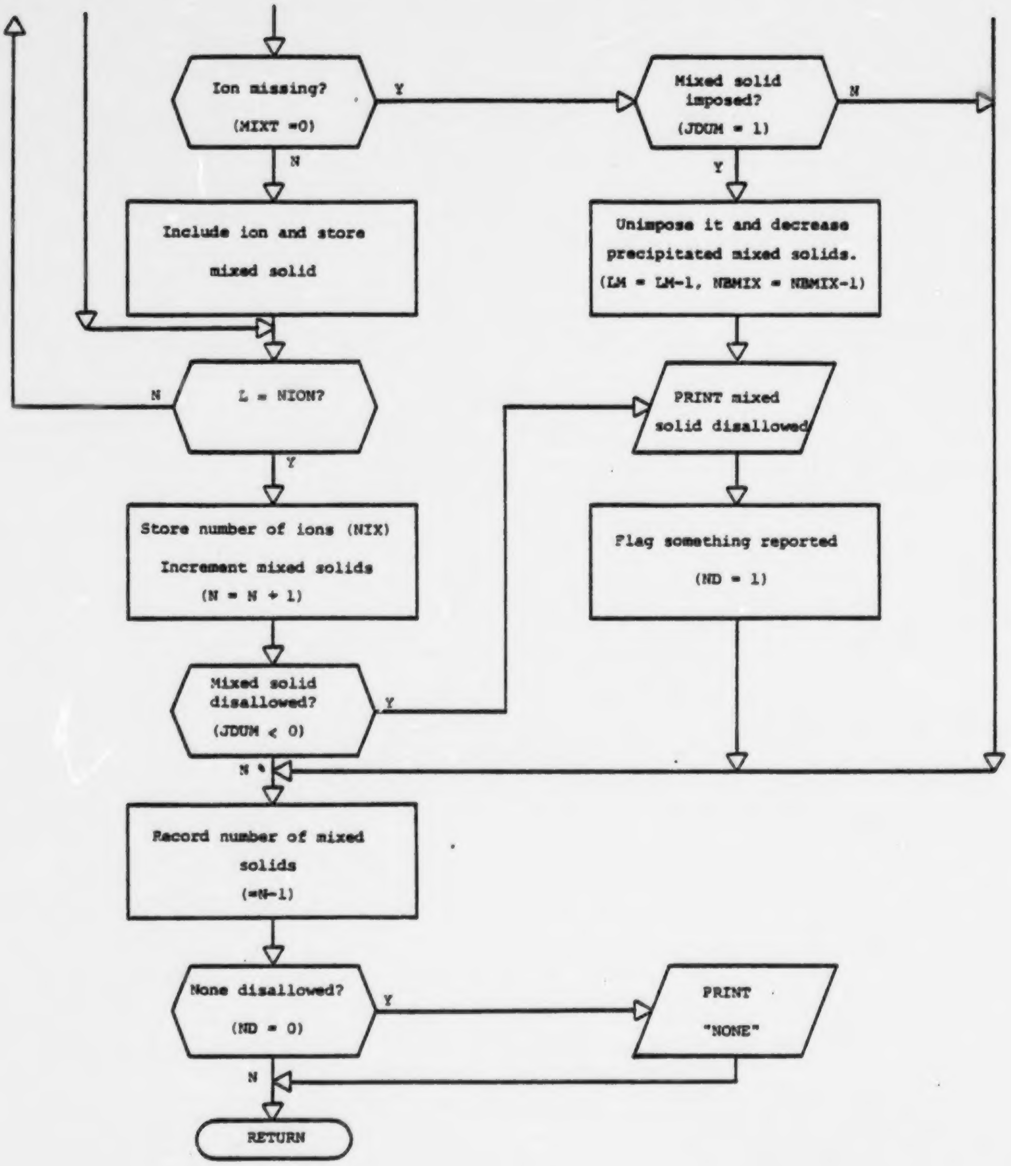
MIX(I,J)	Variable number of J^{th} ion
NIX(I)	Number of ions in I^{th} mixed solid
KSMIX(I,J)	Stoichiometry of J^{th} ion in I^{th} mixed solid
KSHIX(I)	Stoichiometry of hydrogen in mixed solid I
NAMEM(I)	Name
SOLIX(I)	Formation constant (-log) for solid I

Upon exit from INMIX, the variable NMIX is the count of mixed solids which are considered in the system.


INMIX
Number of mixed solids
(NTDF)
TDF
None?
(NTDF = 0)
Y
N
Mixed solid wanted?
NBMIX ≠ 0)
N
Y
Skip in TDF
CALL SKIP
No mixed solids
(NMIX = 0)
RETURN
Too many mixed solids?
(NTDF > 60)
Y
N
CALL SERROR
Status of mixed solids
JDUM = 0
=-1 if all disallowed
(NBMIX =3)
Any imposed or disallowed mixed solids?
(NBMIX = 2)
N
Y
No precipitated mixed solids
Set NBMIX = 0
Mixed solids imposed or disallowed
UDF
Wrong card?
Y
N
CALL SERRIN

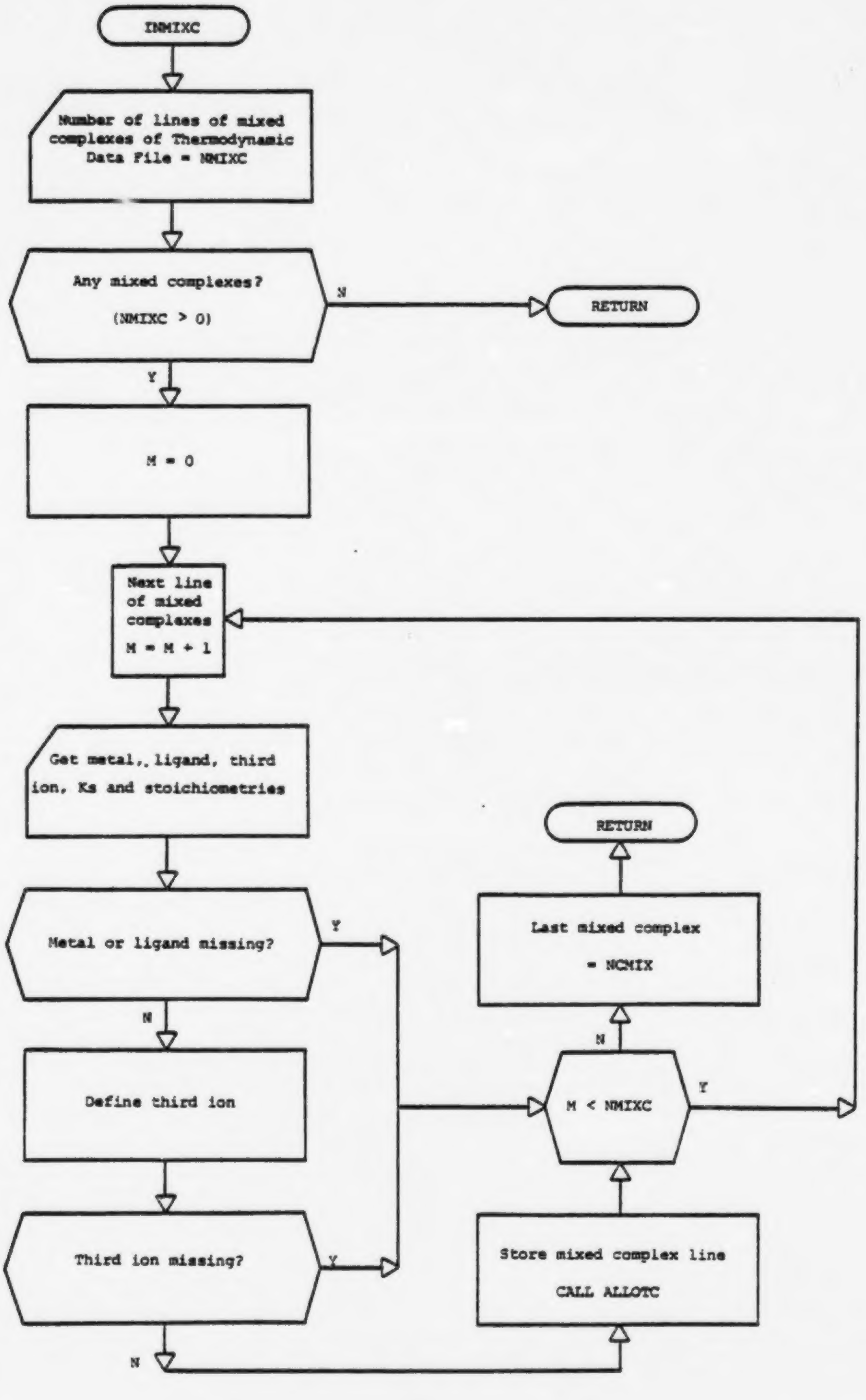

K = 0
K = K + 1
Status?
(IDLM)
= 0
0
< 0
Impose mixed solid
(JDUM = 1)
Disallow mixed solid
(JDUM = -1)
Increment precipitated mixed solid
(NBMIX = NBMIX + 1)
N
K = 19?
Y
Define ions 5 and 6
Initialize counters
N = 1, LM = 0
Data?
NTDF ≠ 0)
N
No precipitated mixed solids
(NBMIX = 0)
Y
I = 0
Next mixed solid
I = I + 1


Number, type, stoichiometry, stability constant
UDF
Too many mixed solids?
(N > 10)
Y
N
CALL XERROR
Imposed?
(JDCM = 1)
Y
N
Increment precipitated mixed solids (LM = LM + 1)
Associate mixed solid with precipitated mixed solid
(MSOLID)
Define ions 1 and 2
as metals
Clay?
NC ≠ 0)
Y
N
Define ions 3 and 4
as metals
Total ions = 6
(NIONS)
Define ions 3 and 4
as ligands
Total ions = 4
(NIONS)
L = 0
Next ion
L = L + 1
No
stoichiometry?
Y
N


Ion missing?
(MIXT =0)
Y
N
Mixed solid imposed?
(JDUM = 1)
N
Y
Include ion and store mixed solid
Unimpose it and decrease precipitated mixed solids.
(LM = LM-1, NBMIX = NBMIX-1)
N
L = NION?
Y
PRINT mixed solid disallowed
Store number of ions (NIX)
Increment mixed solids
(N = N + 1)
Flag something reported
(ND = 1)
Mixed solid disallowed?
(JDUM < 0)
Y
N
Record number of mixed solids
(=N-1)
None disallowed?
(ND = 0)
Y
PRINT
"NONE"
N
RETURN

INMIXC

Subroutine INMIXC is called by SUB1 to perform input for mixed complexes, a mixed complex being a soluble complex which contains a third ion other than H^+ or OH^-.


INMIXC
Number of lines of mixed complexes of Thermodynamic Data File = NMIXC
Any mixed complexes?
(NMIXC > 0)
N
RETURN
Y
M = 0
Next line of mixed complexes
M = M + 1
Get metal, ligand, third ion, Ks and stoichiometries
Metal or ligand missing?
Y
N
Define third ion
Third ion missing?
Y
N
RETURN
Last mixed complex
= NCMIX
N
M < NMIXC
Y
Store mixed complex line
CALL ALLOTC

INRED

Subroutine INRED is used by SUB1 to perform input of data for redox reactions from the user card deck and the Thermodynamic Data File. Redox solids are stored as in ALLOTS. Additionally:

MSRED = Number of redox solids

NELS(I) = Stoichiometry of electrons in the redox solid

Ion-ion redoxes require all new definition:

MREDOX(I) = Variable number of reactant ion

LREDOX(I) = Variable number of product ion

KMR(I) = Stoichiometry of reactant ion

KLR(I) = Stoichiometry of product ion

NHRED(I) = Number of hydrogen consumed

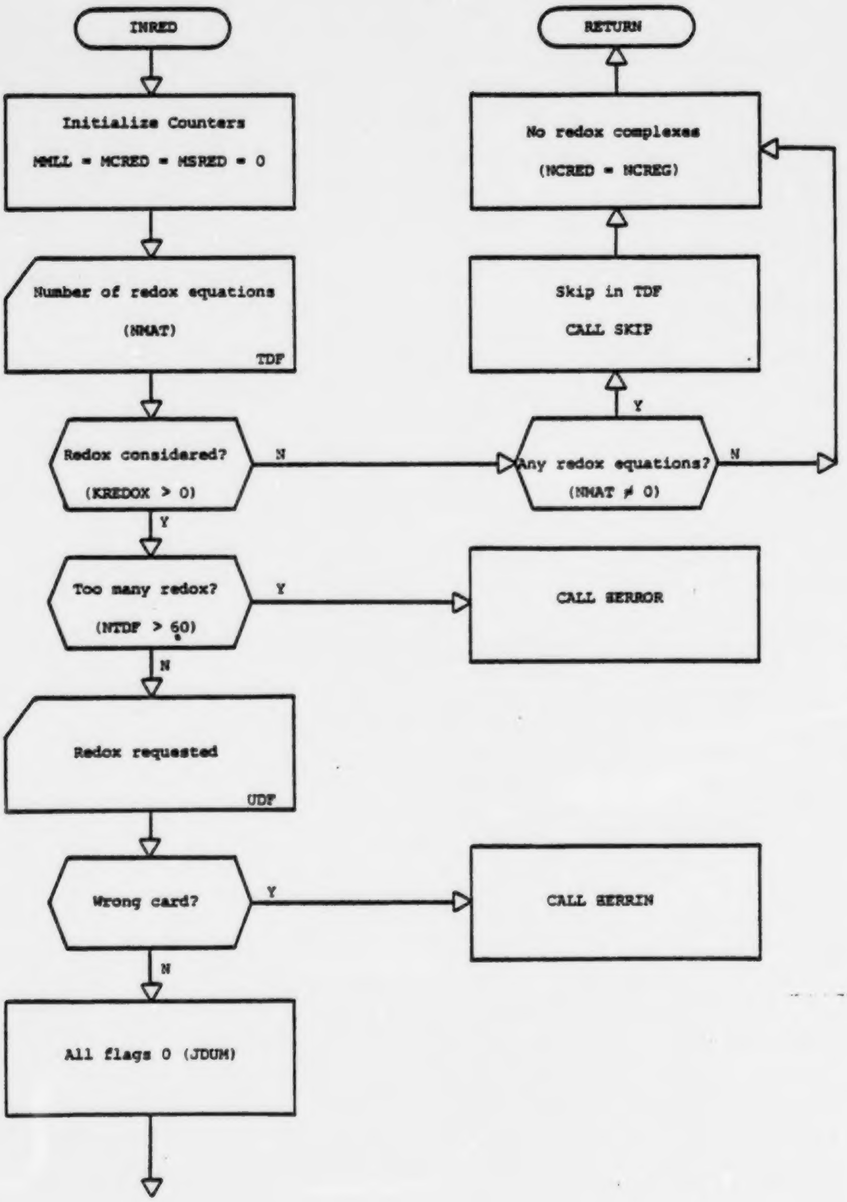
NELEC(I) = Number of electrons produced

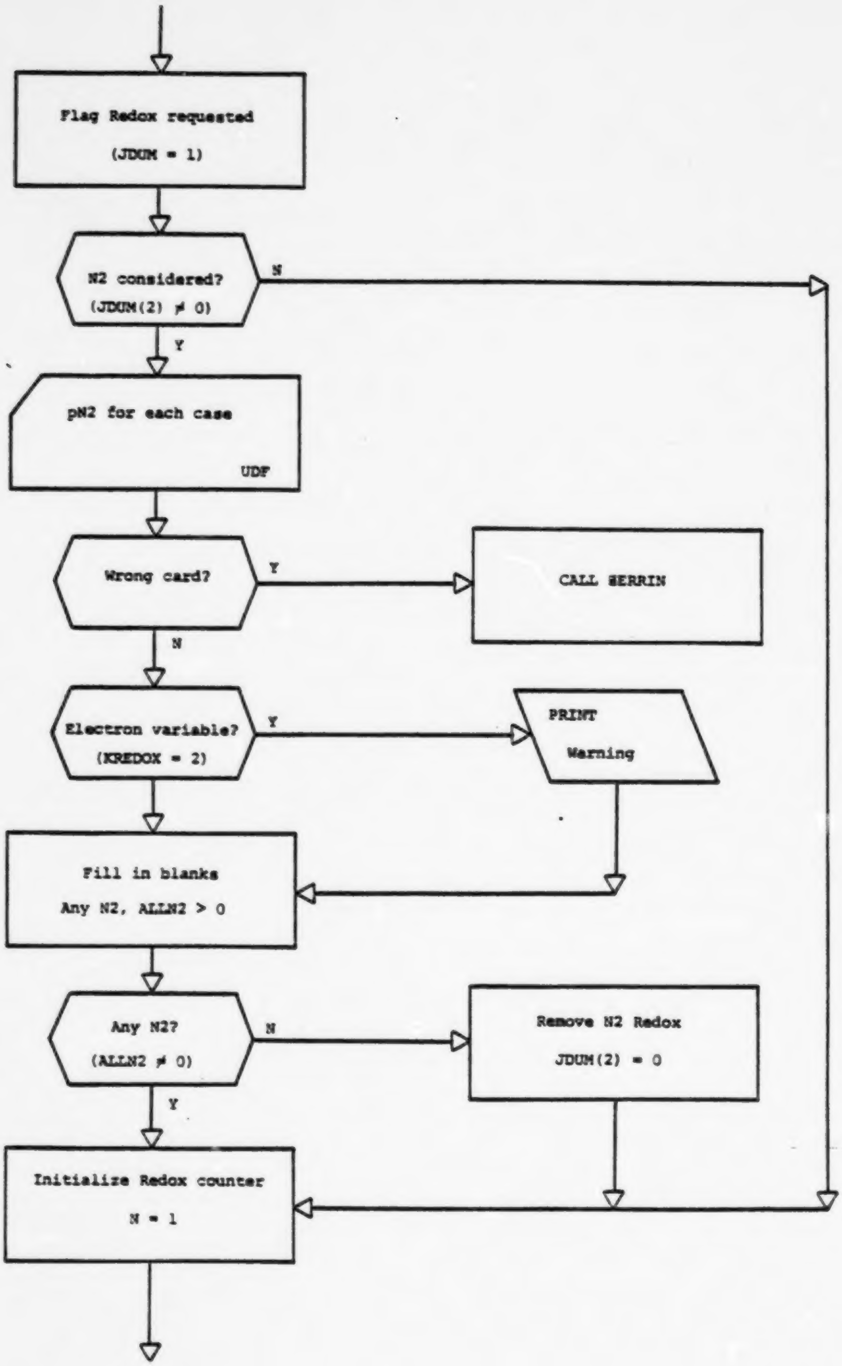
REDCST(I) = The reaction constant (-log)

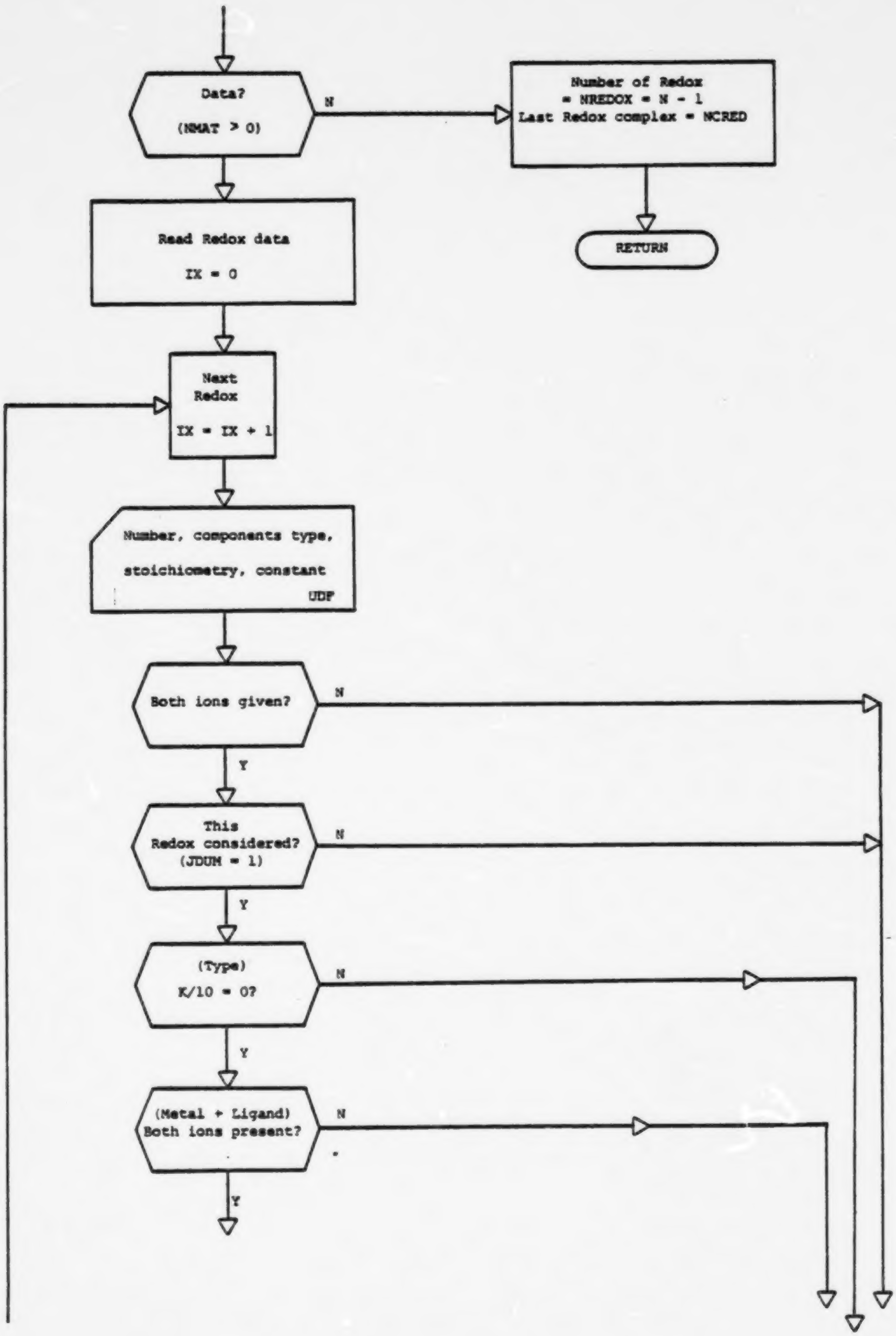
MMLL = Number of ion-ion redoxes

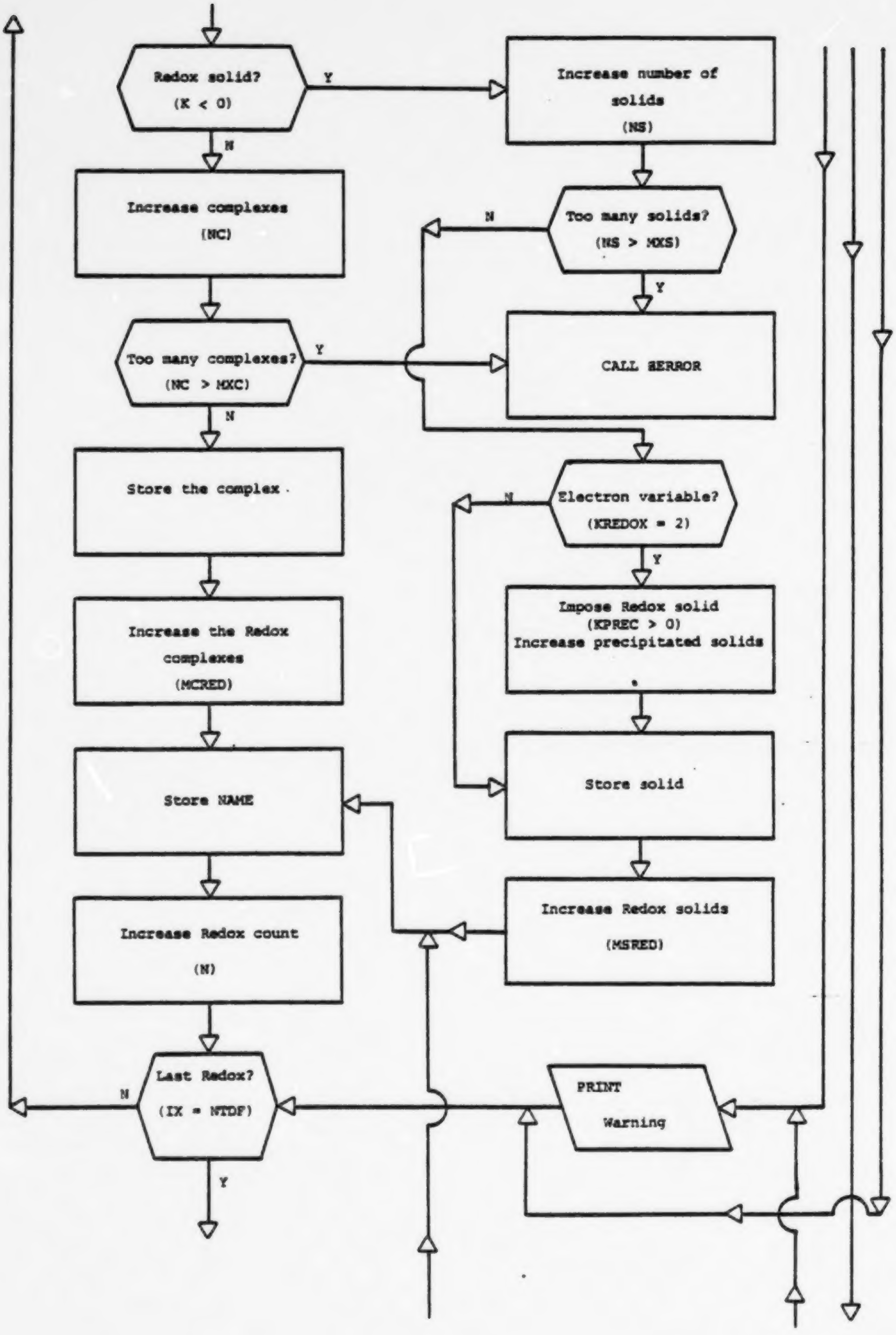
For all redoxes:

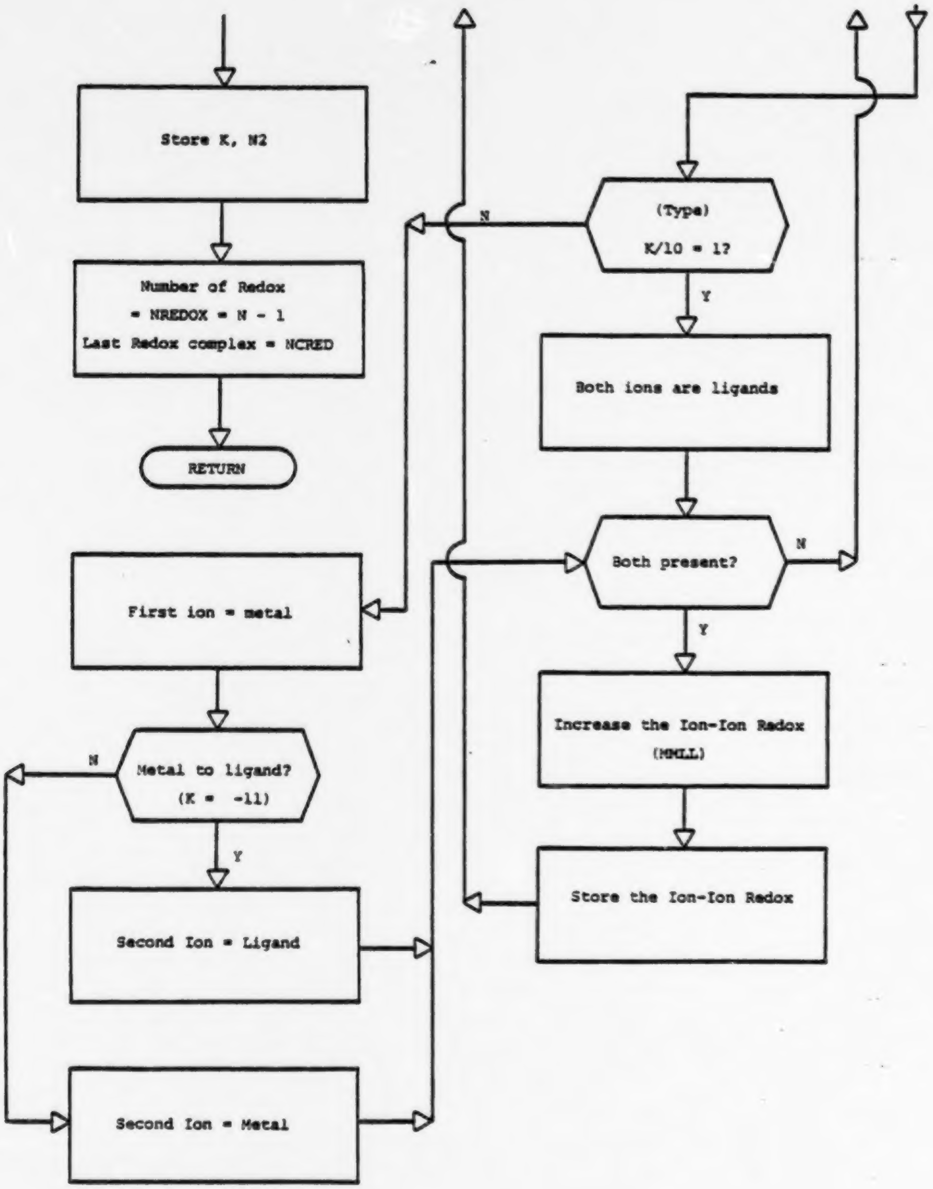
NAMER(J) = Name of the J^{th} redox reaction


INRED
Initialize Counters
MMLL = MCRED = MSRED = 0
Number of redox equations
(NMAT)
TDF
Redox considered?
(KREDOX > 0)
N
Y
Too many redox?
(NTDF > 60)
Y
N
CALL SERROR
Redox requested
UDF
Wrong card?
Y
N
CALL SERRIN
All flags 0 (JDUM)
RETURN
No redox complexes
(NCRED = NCREG)
Skip in TDF
CALL SKIP
Y
Any redox equations?
(NMAT ≠ 0)
N


Flag Redox requested
(JDUM = 1)
N2 considered?
(JDUM(2) ≠ 0)
N
Y
pN2 for each case
UDF
Wrong card?
Y
CALL $ERRIN
N
Electron variable?
(KREDOX = 2)
Y
PRINT
Warning
Fill in blanks
Any N2, ALLN2 > 0
Any N2?
(ALLN2 ≠ 0)
N
Remove N2 Redox
JDUM(2) = 0
Y
Initialize Redox counter
N = 1

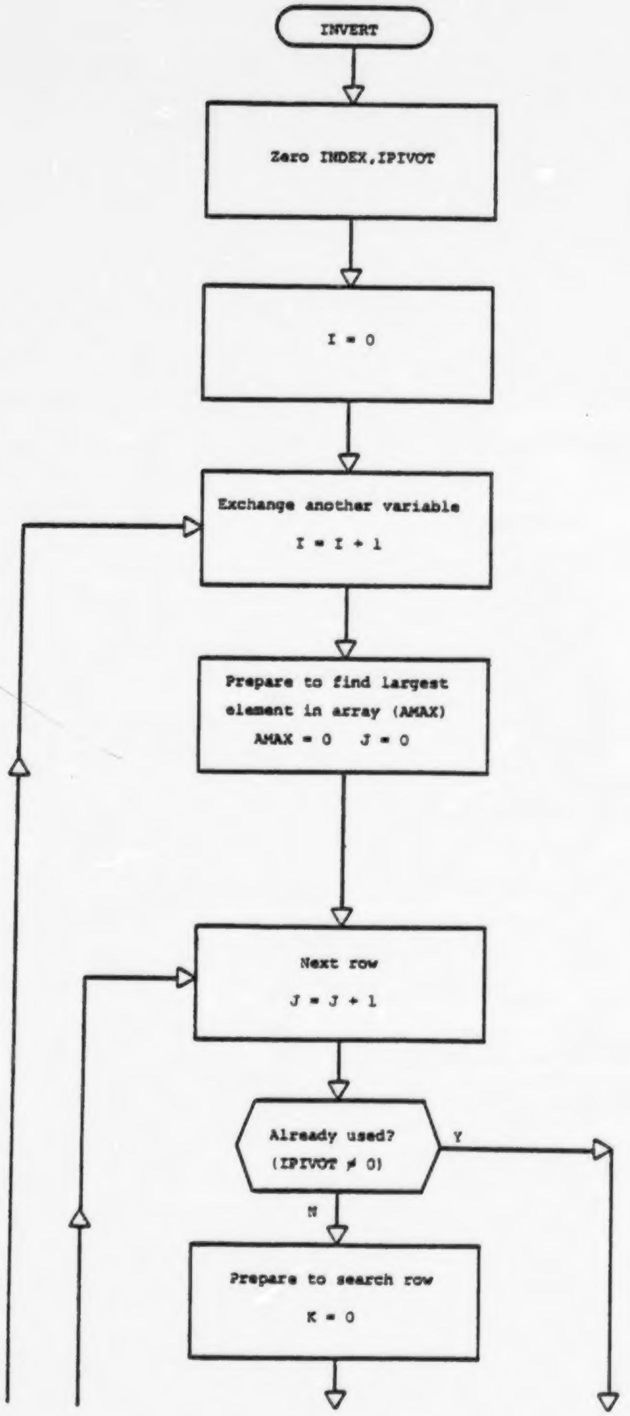

Data?
(NMAT > 0)
N
Number of Redox
= NREDOX = N - 1
Last Redox complex = NCRED
RETURN
Read Redox data
IX = 0
Next
Redox
IX = IX + 1
Number, components type,
stoichiometry, constant
UDF
Both ions given?
N
Y
This
Redox considered?
(JDUM = 1)
N
Y
(Type)
K/10 = 0?
N
Y
(Metal + Ligand)
Both ions present?
N
Y

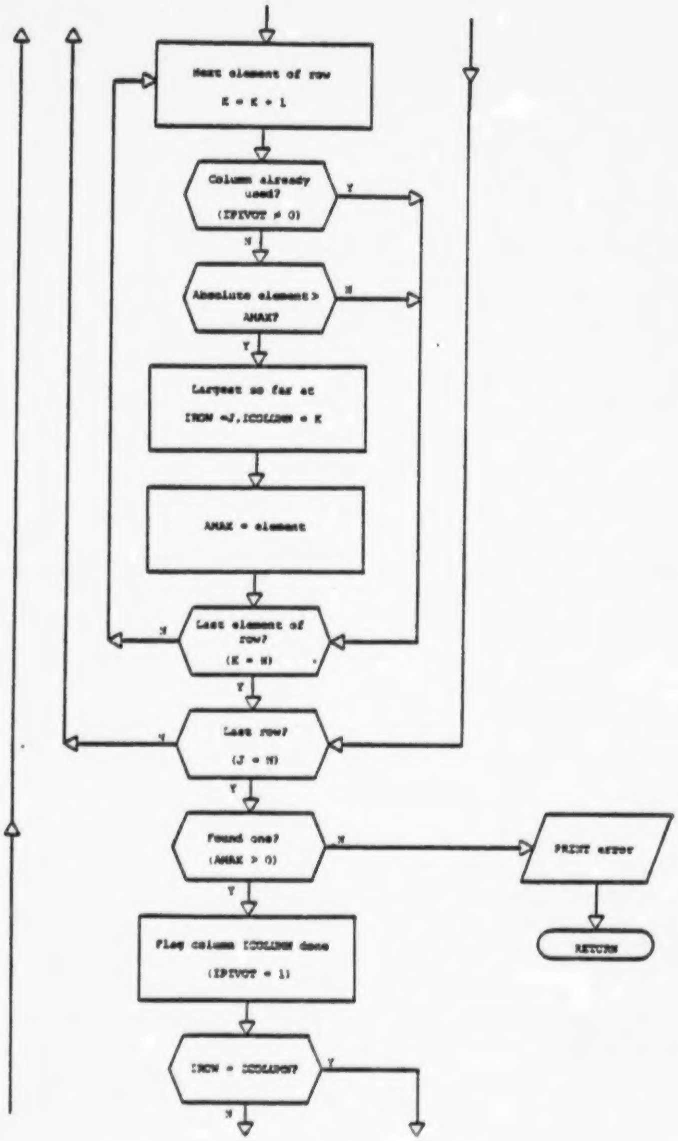

Redox solid?
(K < 0)
Y
N
Increase number of solids
(NS)
Increase complexes
(NC)
N
Too many solids?
(NS > MXS)
Y
Too many complexes?
(NC > MXC)
Y
N
CALL ZERROR
Store the complex
N
Electron variable?
(KREDOX = 2)
Y
Impose Redox solid
(KPREC > 0)
Increase precipitated solids
Increase the Redox complexes
(MCRED)
Store NAME
Store solid
Increase Redox count
(N)
Increase Redox solids
(MSRED)
Last Redox?
(IX = NTDF)
N
Y
PRINT
Warning

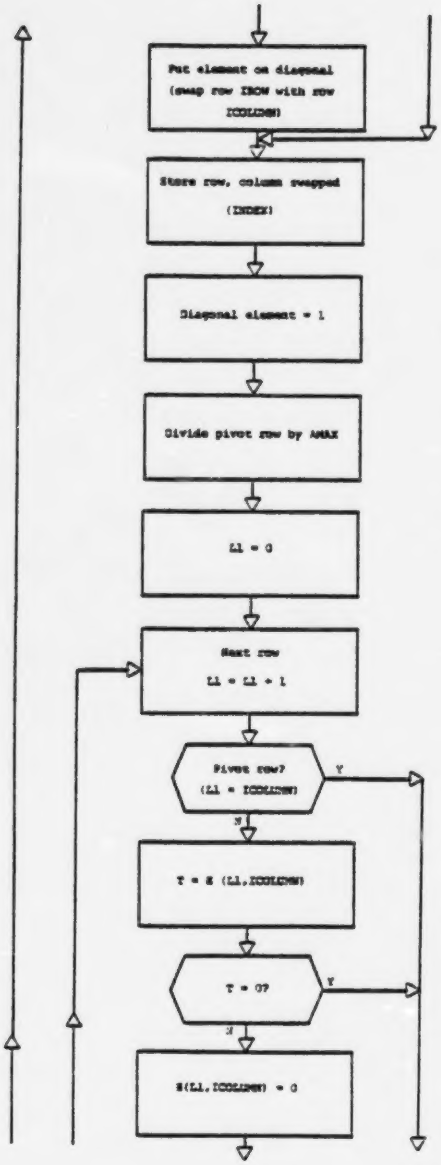

Store K, N2
Number of Redox
= NREDOX = N - 1
Last Redox complex = NCRED
RETURN
First ion = metal
Metal to ligand?
(K = -11)
N
Y
Second Ion = Ligand
Second Ion = Metal
(Type)
K/10 = 1?
N
Y
Both ions are ligands
Both present?
N
Y
Increase the Ion-Ion Redox
(MMLL)
Store the Ion-Ion Redox

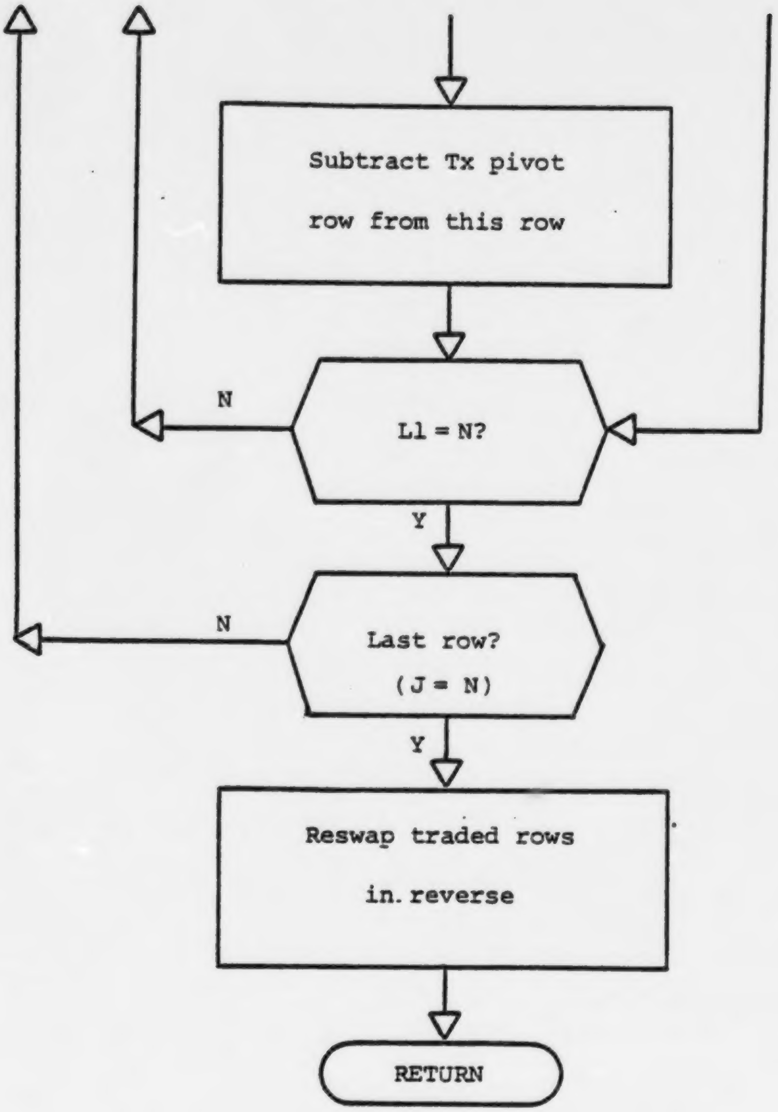
INVERT

INVERT is called by SUBJAC to calculate $[Z]^{-1}$ of the NIQxNIQ matrix Z.


INVERT
Zero INDEX,IPIVOT
I = 0
Exchange another variable
I = I + 1
Prepare to find largest
element in array (AMAX)
AMAX = 0   J = 0
Next row
J = J + 1
Already used?
(IPIVOT ≠ 0)
Y
N
Prepare to search row
K = 0

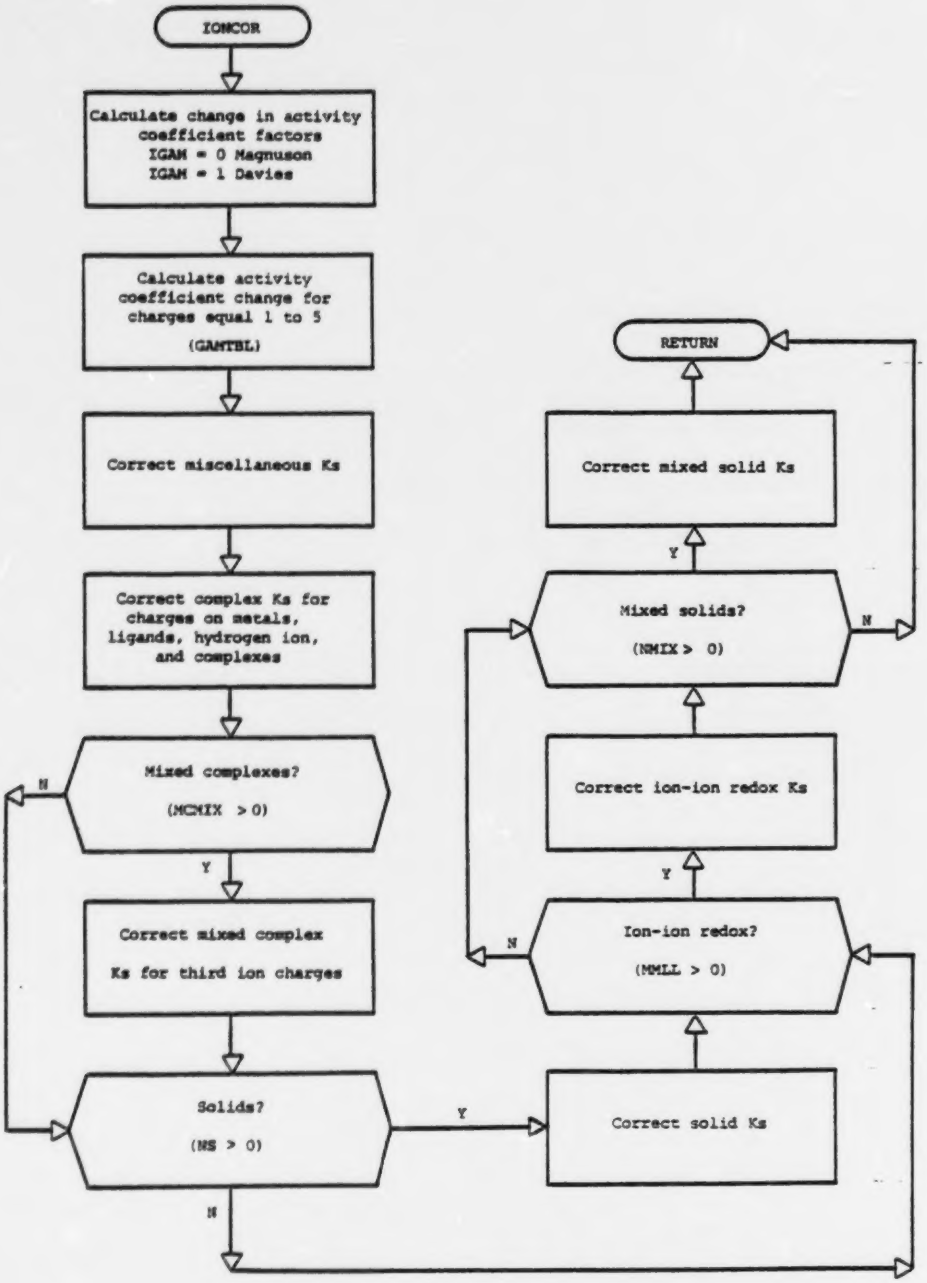

Next element of row
K = K + 1
Column already used?
(IPIVOT ≠ 0)
Y
N
Absolute element > AMAX?
N
Y
Largest so far at
IROW =J, ICOLUMN = K
AMAX = element
Last element of row?
(K = N)
N
Y
Last row?
(J = N)
N
Y
Found one?
(AMAX > 0)
N
PRINT error
RETURN
Y
Flag column ICOLUMN done
(IPIVOT = 1)
IROW = ICOLUMN?
Y
N


Put element on diagonal (swap row IROW with row ICOLUMN)
Store row, column swapped (INDEX)
Diagonal element = 1
Divide pivot row by AMAX
L1 = 0
Next row
L1 = L1 + 1
Pivot row? (L1 = ICOLUMN)
Y
N
T = A (L1,ICOLUMN)
T = 0?
Y
N
A(L1,ICOLUMN) = 0


Subtract Tx pivot row from this row
L1 = N?
N
Y
Last row? (J = N)
N
Y
Reswap traded rows in. reverse
RETURN

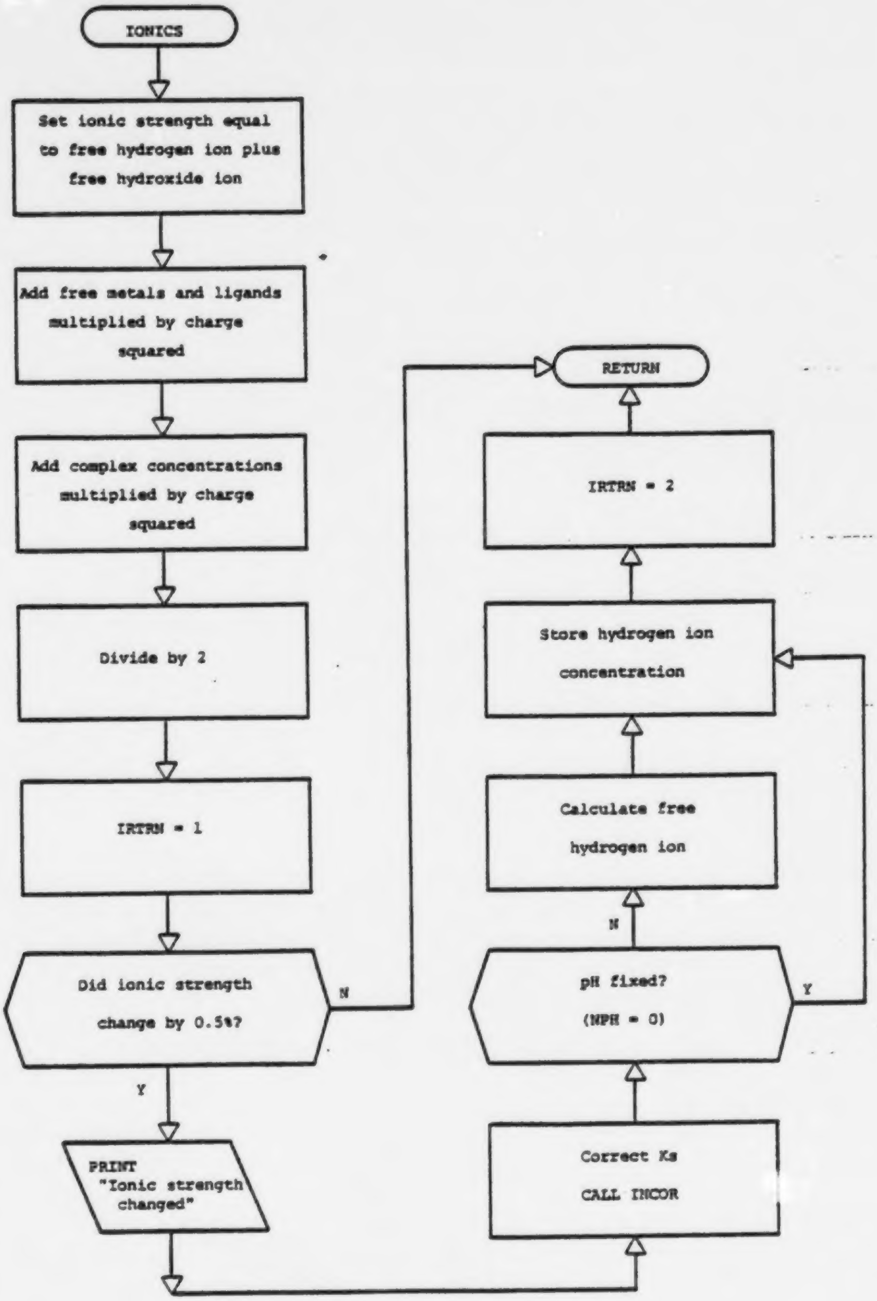
IONCOR

Subroutine IONCOR is called by SUB2 if the original ionic strength is fixed and by IONICS after the latter has made sufficient alteration in the value of the ionic strength of the system to warrant correction of formation constants to reflect the new ionic strength.


IONCOR
Calculate change in activity coefficient factors
IGAM = 0 Magnuson
IGAM = 1 Davies
Calculate activity coefficient change for charges equal 1 to 5
(GAMTBL)
Correct miscellaneous Ks
Correct complex Ks for charges on metals, ligands, hydrogen ion, and complexes
Mixed complexes?
(MCMIX > 0)
N
Y
Correct mixed complex Ks for third ion charges
Solids?
(NS > 0)
N
Y
Correct solid Ks
Ion-ion redox?
(MMLL > 0)
N
Y
Correct ion-ion redox Ks
Mixed solids?
(NMIX > 0)
N
Y
Correct mixed solid Ks
RETURN

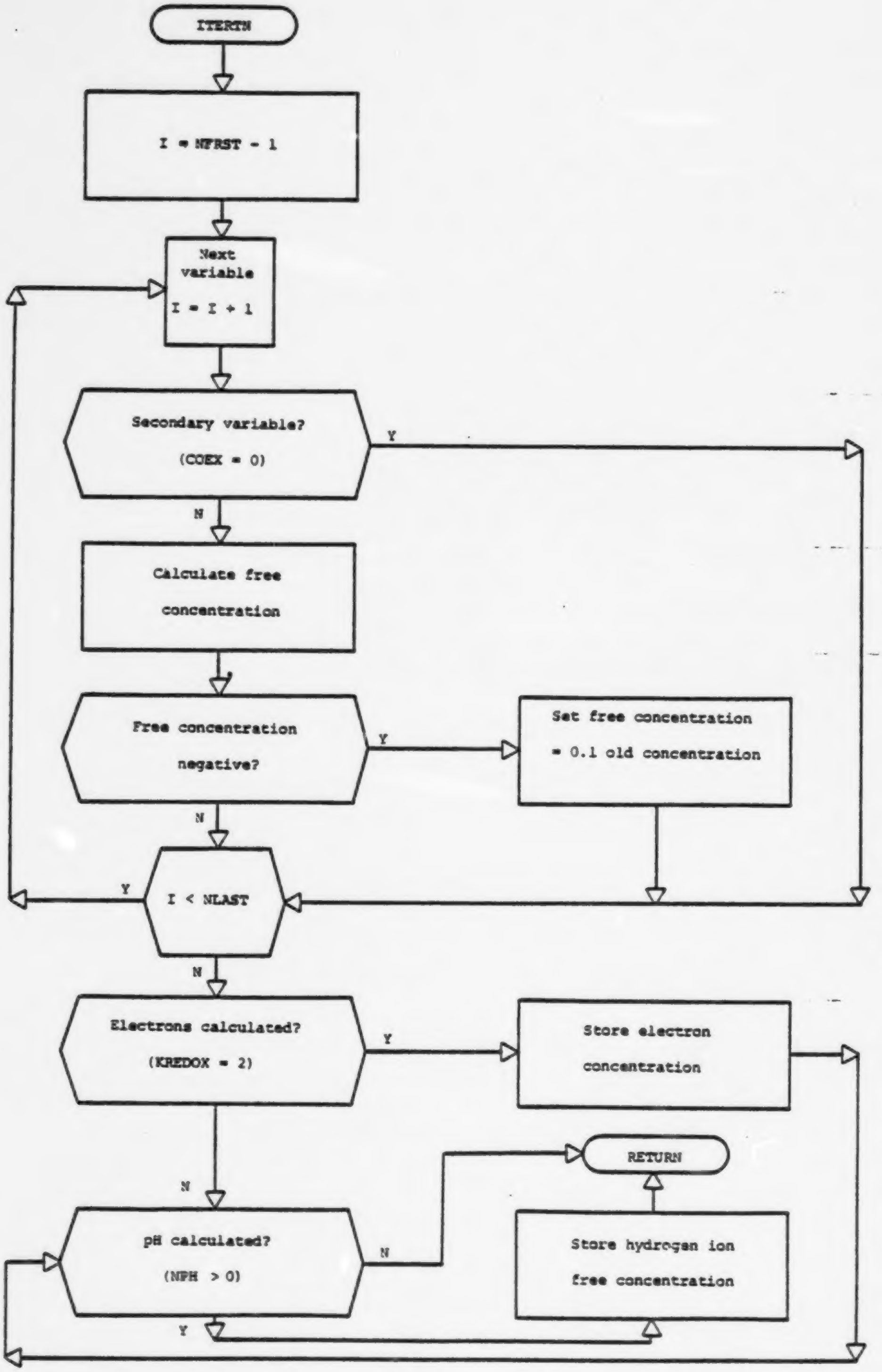
IONICS

Subroutine IONICS is called by SUB3 to calculate a new ionic strength (STRNGH) for the system, and if there is a significant change, IONCOR is called to make corrections to formation constants.


IONICS
Set ionic strength equal to free hydrogen ion plus free hydroxide ion
Add free metals and ligands multiplied by charge squared
Add complex concentrations multiplied by charge squared
Divide by 2
IRTRN = 1
Did ionic strength change by 0.5%?
N
Y
PRINT "Ionic strength changed"
Correct Ks CALL INCOR
pH fixed? (NPH = 0)
N
Y
Calculate free hydrogen ion
Store hydrogen ion concentration
IRTRN = 2
RETURN

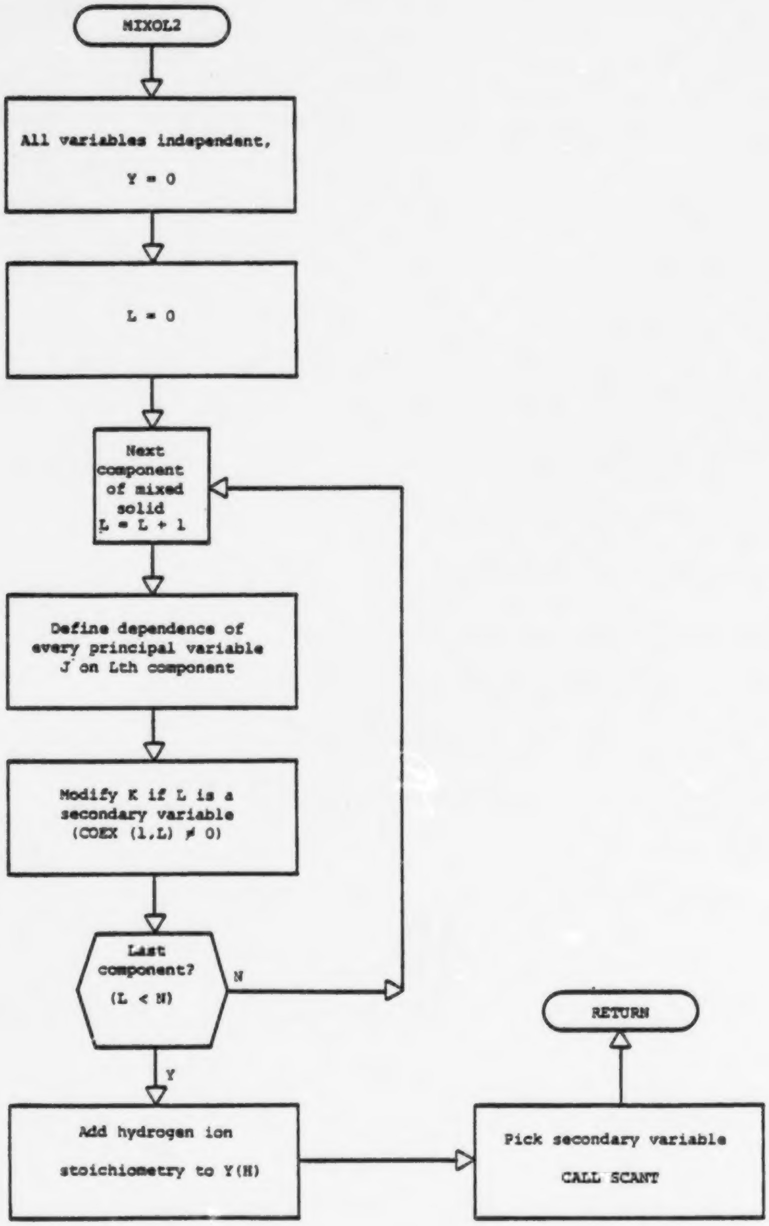
ITERTN

Subroutine ITERTN is called by SUB3 to add the free concentration adjustments, Y(I), computed in GAUSSL, to the array of free concentrations (EXM) or to reduce the free concentration to 1/10 of its previous value if the correction would cause it to become negative.


ITERTN
I = NFRST - 1
Next variable
I = I + 1
Secondary variable?
(COEX = 0)
Y
N
Calculate free concentration
Free concentration negative?
Y
Set free concentration = 0.1 old concentration
N
I < NLAST
Y
N
Electrons calculated?
(KREDOX = 2)
Y
Store electron concentration
N
RETURN
pH calculated?
(NPH > 0)
N
Store hydrogen ion free concentration
Y

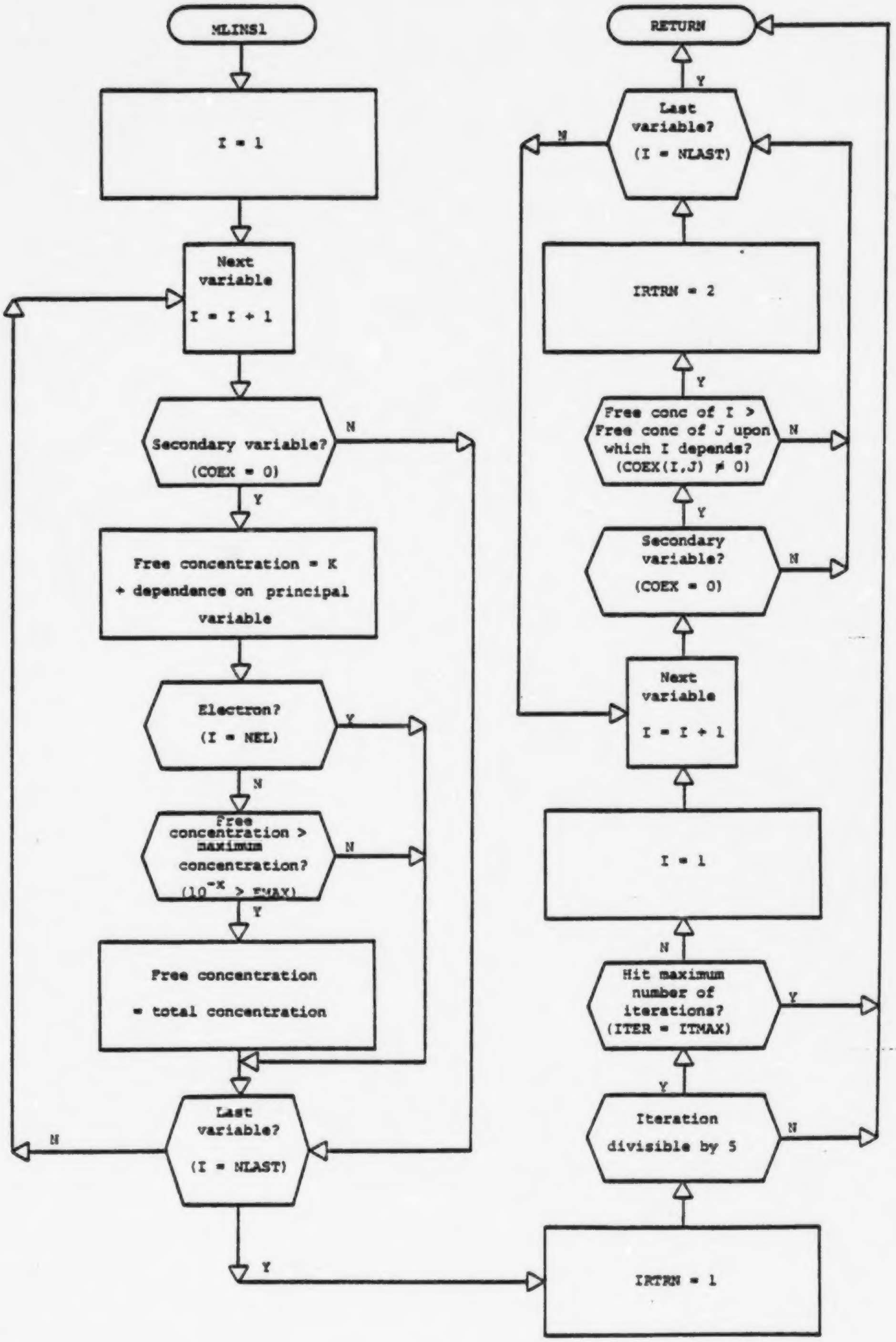
MIXOL2

MIXOL2 is used by SECVAR and CHKSOL to define a new secondary variable due to an imposed mixed solid (if called by SECVAR) or to precipitation of a mixed solid (if called by CHKSOL). The position of the solid which is being imposed or precipitated is transmitted to MIXOL2 via the parameter K. Any principal variable upon which a component of the new solid depends is a candidate for the new secondary variable. Candidates are flagged by setting Y for the variable to a value different from zero. If any components of the mixed solid are already secondary variables, the formation constant is modified. After all candidates have been flagged, SCAN1 is called to make the final selection of the new secondary variable.


MIXOL2
All variables independent,
Y = 0
L = 0
Next component of mixed solid
L = L + 1
Define dependence of every principal variable J on Lth component
Modify K if L is a secondary variable (COEX (1,L) ≠ 0)
Last component?
(L < N)
N
Y
Add hydrogen ion stoichiometry to Y(H)
Pick secondary variable
CALL SCANT
RETURN

MLINS1

MLINS1 is used by SUB3 at the beginning of the loop which attempts to attain convergence to compute the concentration of any secondary variables. These secondary variable concentrations are computed as the -log of the solubility product plus the free concentration of each related principal variable. Since the free concentration obviously cannot exceed the possible total concentration of an ion, should this result the free concentration is reset to the possible total concentration. On every fifth call to MLINS1, a test is made to ascertain that the free concentrations of secondary variables are less than those of the principal variables upon which they depend; in the event that one or more is not less, IRTRN is set to 2 to signal SUB3 to call SECVAR to reselect secondary variables.


MLINS1
I = 1
Next variable
I = I + 1
Secondary variable?
(COEX = 0)
N
Y
Free concentration = K
+ dependence on principal variable
Electron?
(I = NEL)
Y
N
Free concentration > maximum concentration?
(10^-X > TMAX)
N
Y
Free concentration
= total concentration
Last variable?
(I = NLAST)
N
Y
IRTRN = 1
Iteration divisible by 5
N
Y
Hit maximum number of iterations?
(ITER = ITMAX)
Y
N
I = 1
Next variable
I = I + 1
Secondary variable?
(COEX = 0)
N
Y
Free conc of I > Free conc of J upon which I depends?
(COEX(I,J) ≠ 0)
N
Y
IRTRN = 2
Last variable?
(I = NLAST)
N
Y
RETURN

MODIX

MODIX is used to modify the error vector (Y) and the Jacobian matrix (Z) to reflect the presence of secondary variables. It is called by SUB3 after computation of the Z matrix and the Y error vector if secondary variables exist (N2V>0). It is also called by SUBJAC during final computation of the Z matrix prior to printing of interaction intensities and capacities.

Most of the routine is within a single loop which scans the entire COEX matrix (except for H^+): when a secondary variable is found, the derivatives are modified such that:

Z(k,j) = Z(k,j) + Z(k,i) * COEX(i,j) * EXM(i) / EXM(j)

Z(j,k) = Z(j,k) + Z(i,j) * COEX(i,j)

where:

i is the secondary variable

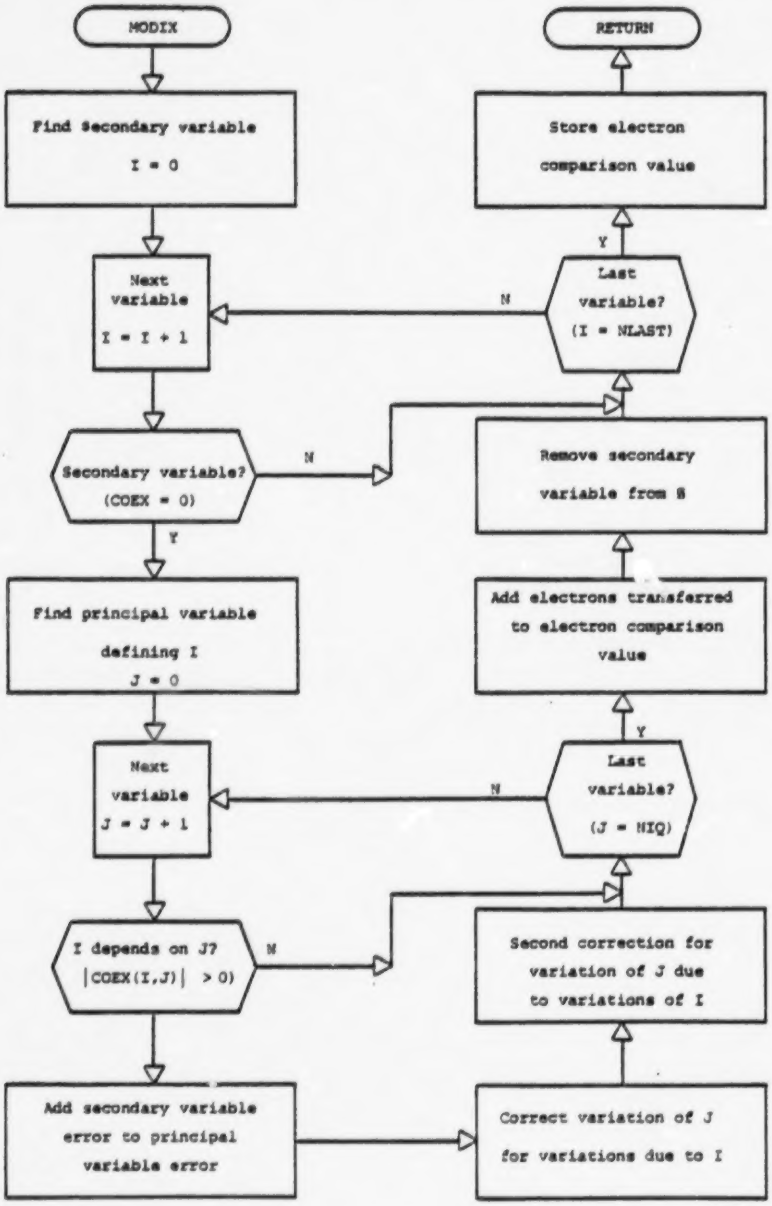
j is the principal variable

for K = 1 to NIQ

and the error vector is modified such that:

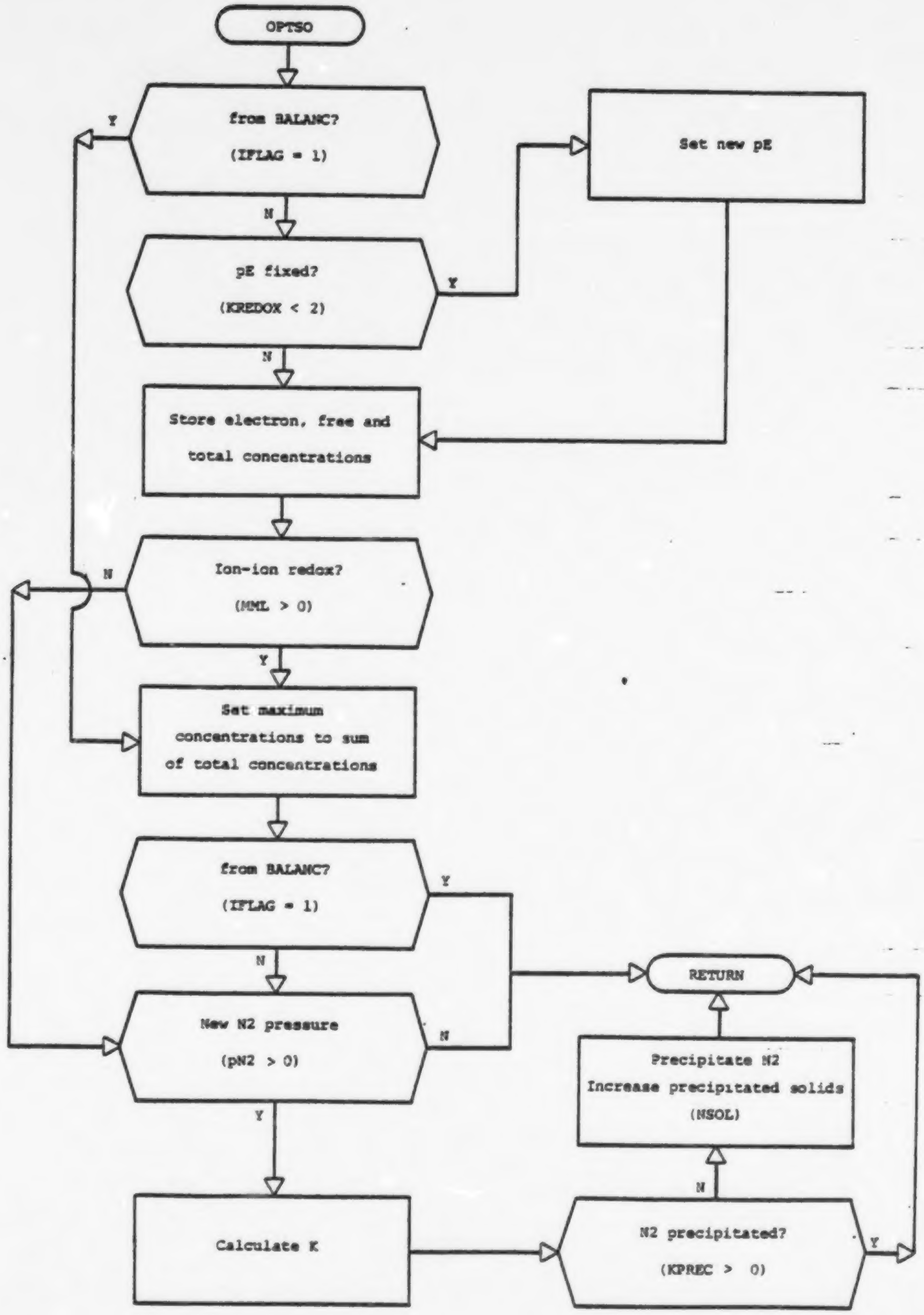
Y(j) = Y(j) + COEX(i,j) * Y(i)

thereby reflecting the effect of any precipitated solid or ion-ion redox. Finally, TESTE is updated to reflect electrons transferred due to redox and the correlation between the secondary variable and associated variables is removed. Prior to RETURN, TESTY for electrons is updated to reflect the accumulated electron transfers temporarily stored in TESTE.


MODIX
Find secondary variable
I = 0
Next variable
I = I + 1
Secondary variable?
(COEX = 0)
N
Y
Find principal variable defining I
J = 0
Next variable
J = J + 1
I depends on J?
|COEX(I,J)| > 0)
N
Add secondary variable error to principal variable error
Correct variation of J for variations due to I
Second correction for variation of J due to variations of I
Last variable?
(J = NIQ)
N
Y
Add electrons transferred to electron comparison value
Remove secondary variable from B
Last variable?
(I = NLAST)
N
Y
Store electron comparison value
RETURN

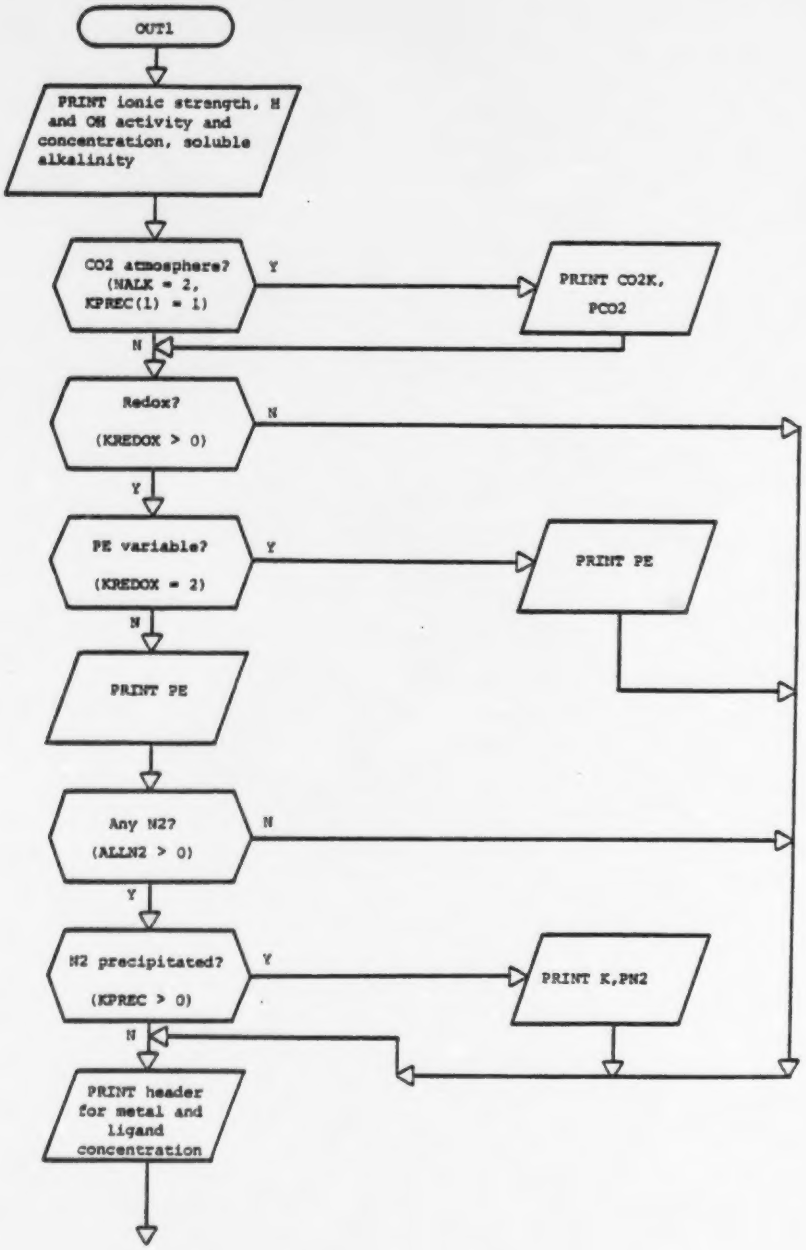
OPTSO

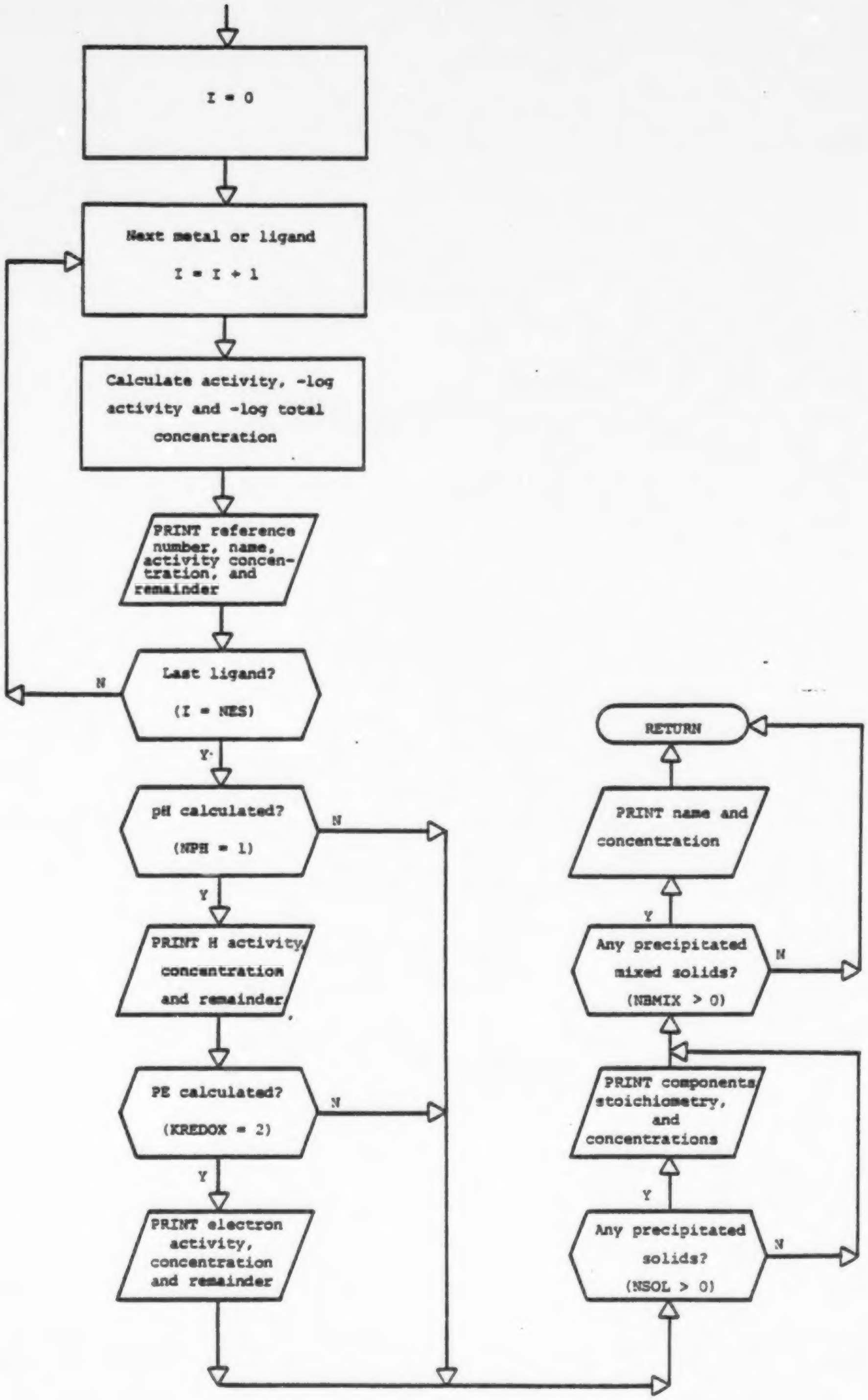
OPTSO is called by SUB3 and BALANC. When called by SUB3, it sets pE for the current case and updates electron activity and other variables associated with pE. If there are ion-ion redox reactions, the maximum total concentrations of all metals and ligands involved in these reactions are determined and stored in EMAX. If pN2 is nonzero, the formation constant for the "solid" N2 is updated and if N2 has not been imposed, it is imposed at this point. When OPTSO is called from BALANC, only maximum concentrations are calculated.


OPTSO
from BALANC?
(IFLAG = 1)
Y
N
Set new pE
pE fixed?
(KREDOX < 2)
Y
N
Store electron, free and total concentrations
Ion-ion redox?
(MML > 0)
N
Y
Set maximum concentrations to sum of total concentrations
from BALANC?
(IFLAG = 1)
Y
N
RETURN
New N2 pressure
(pN2 > 0)
N
Y
Precipitate N2
Increase precipitated solids
(NSOL)
N
Calculate K
N2 precipitated?
(KPREC > 0)
Y

OUT1

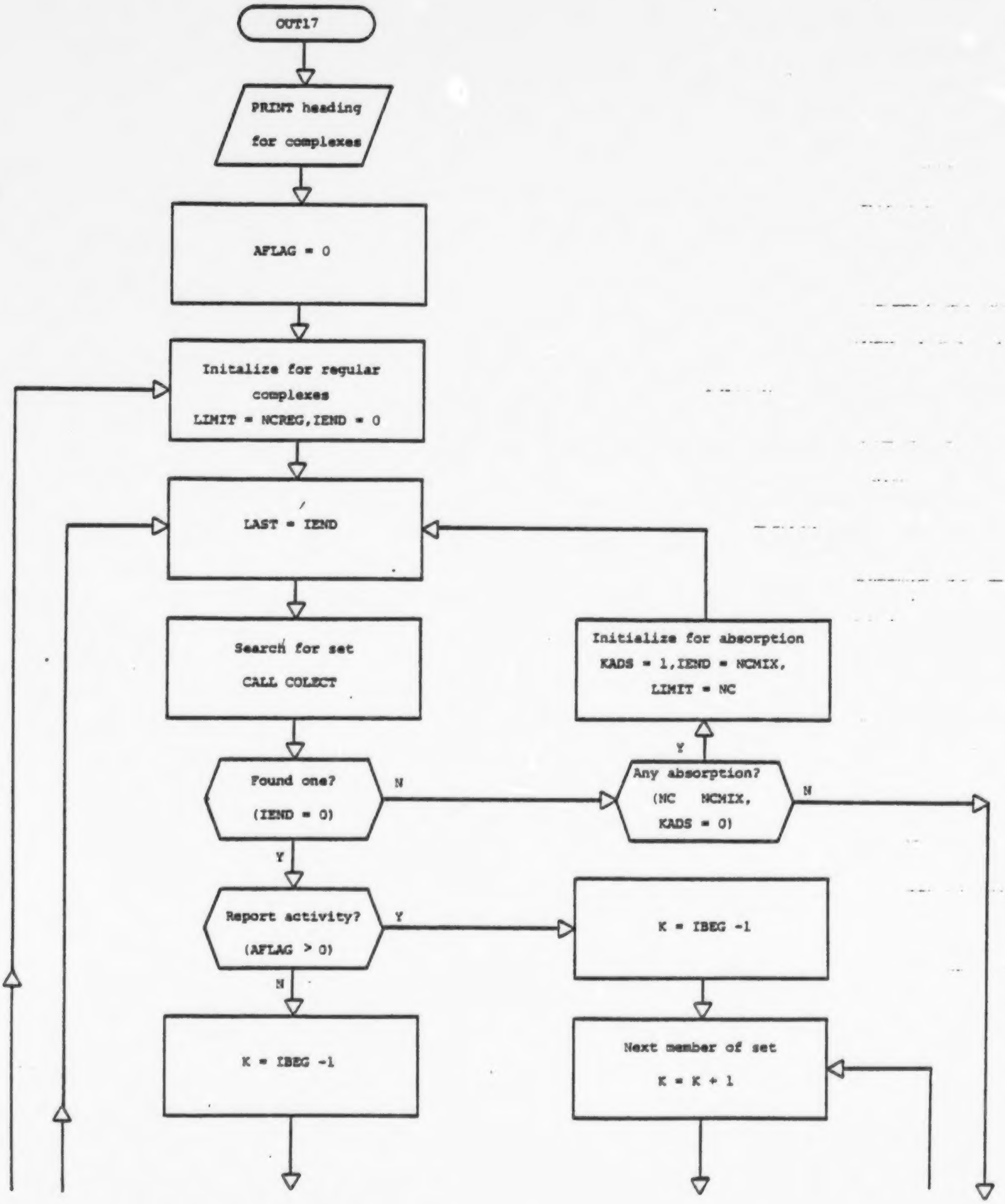
Subroutine OUT1 is called by SUB3 after completion of computations for a case to perform output functions. Ionic strength is reported, followed by pH, pOH, TOTH, the water association constant WK, and H^+ and OH^- concentrations. If CO2 is in the system, PCO2 is reported, followed by electron activity if redox reactions are being considered, then pN2 if N2 is in the system. Then are reported the reference numbers, names, free concentrations, activities (both as moles per liter and as -log molar), and remainders for each ion in the system.

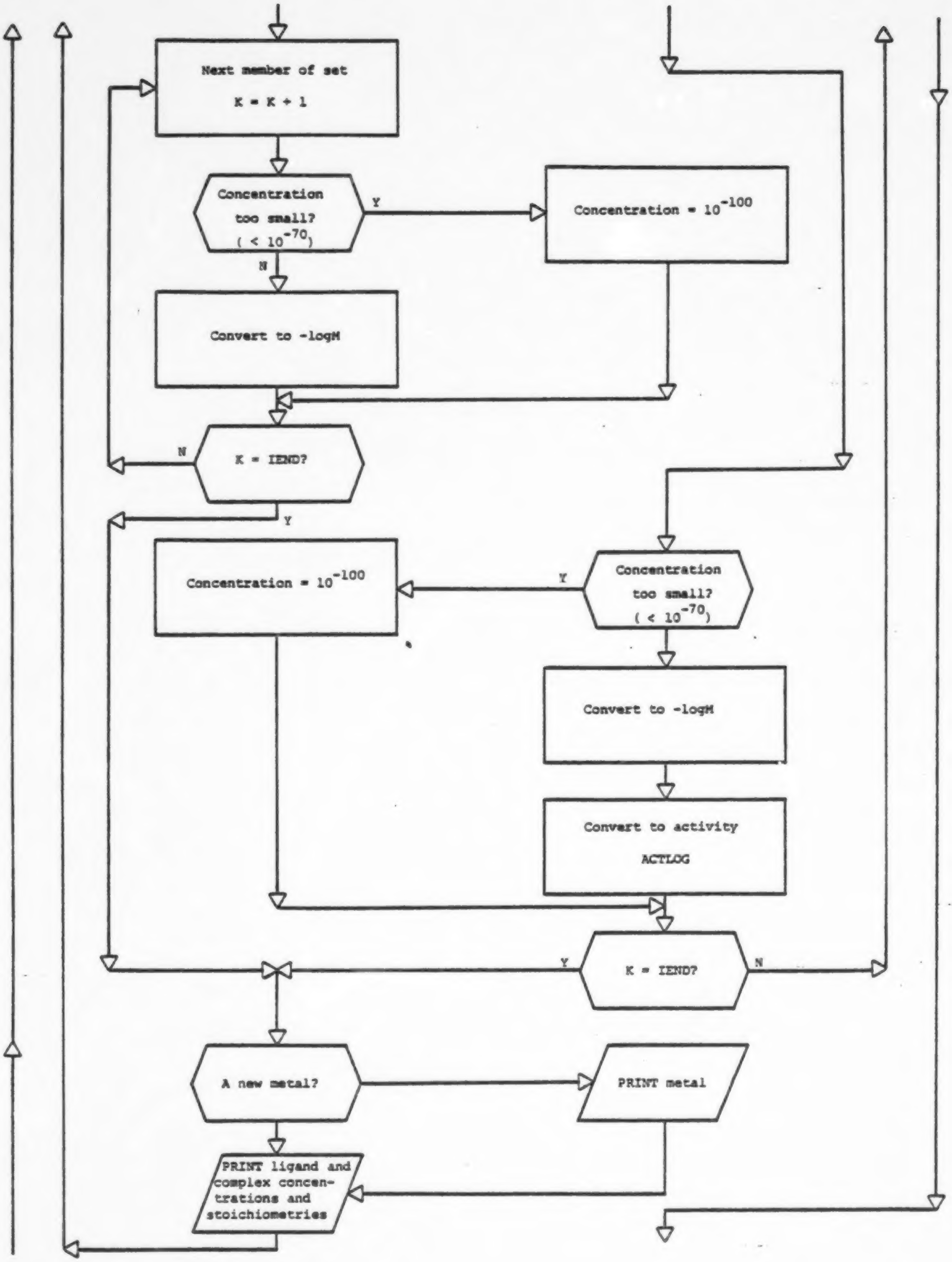

OUT1
PRINT ionic strength, H and OH activity and concentration, soluble alkalinity
CO2 atmosphere? (NALK = 2, KPREC(1) = 1)
Y
PRINT CO2K, PCO2
N
Redox?
(KREDOX > 0)
N
Y
PE variable?
(KREDOX = 2)
Y
PRINT PE
N
PRINT PE
Any N2?
(ALLN2 > 0)
N
Y
N2 precipitated?
(KPREC > 0)
Y
PRINT K,PN2
N
PRINT header for metal and ligand concentration

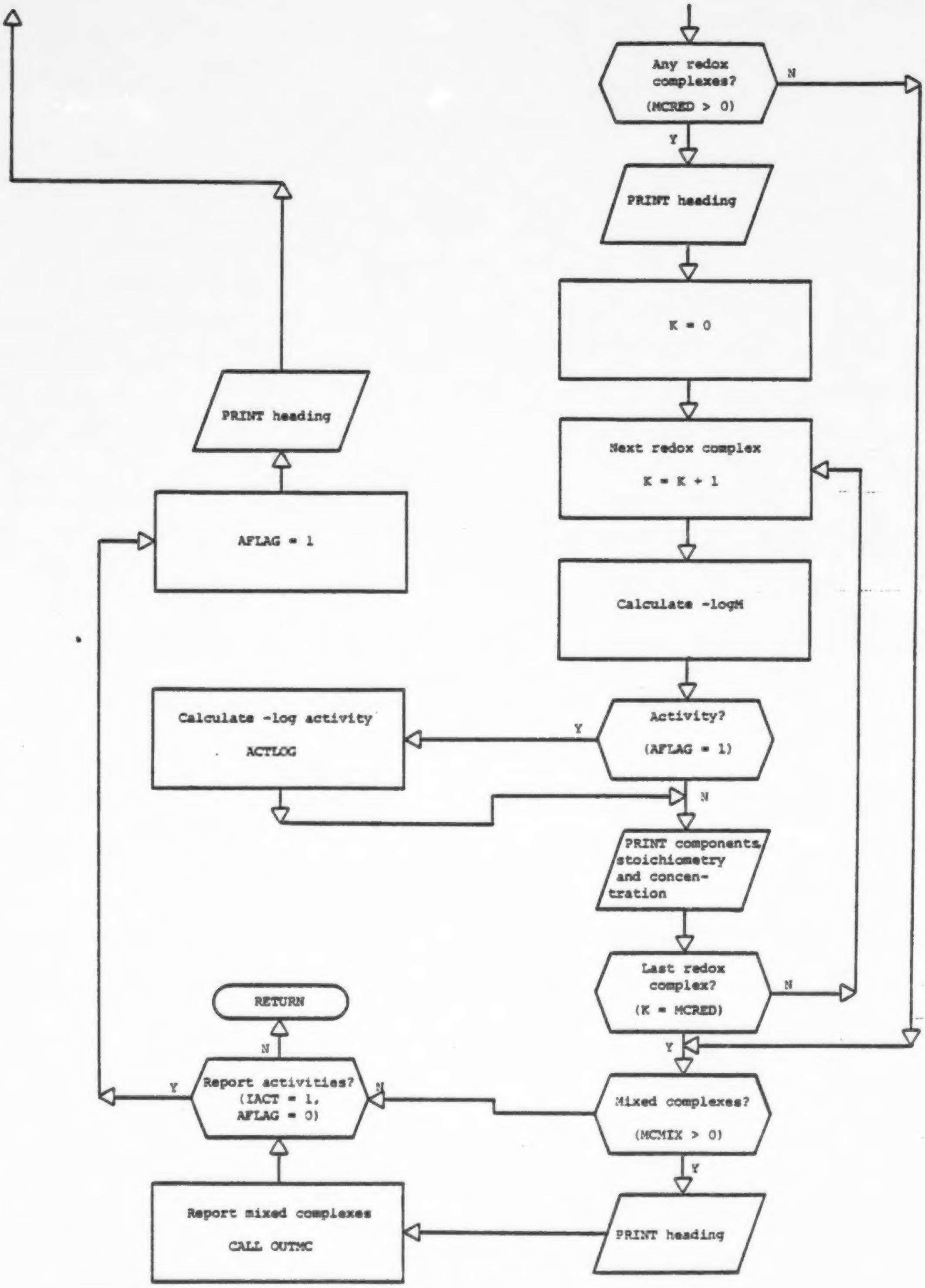

I = 0
Next metal or ligand
I = I + 1
Calculate activity, -log activity and -log total concentration
PRINT reference number, name, activity concentration, and remainder
Last ligand?
(I = NES)
N
Y
pH calculated?
(NPH = 1)
N
Y
PRINT H activity, concentration and remainder
PE calculated?
(KREDOX = 2)
N
Y
PRINT electron activity, concentration and remainder
RETURN
PRINT name and concentration
Y
Any precipitated mixed solids?
(NBMIX > 0)
N
PRINT components, stoichiometry, and concentrations
Y
Any precipitated solids?
(NSOL > 0)
N

OUT17

OUT17 is used by SUB3 to report concentrations of both "regular" and mixed complexes. Concentrations are reported as -log M and may also be reported as -log activities, depending upon the user input value IACT. COLECT is used to find the first and last complexes of each metal-ligand pair and report them. After all regular complexes have been reported, adsorption complexes are reported (partially through use of the same loop). If there are any redox complexes they are reported and then any mixed complexes in the system. If -log activities are desired, the entire routine (well, almost all of it) is re-executed, this time reporting concentrations as -log activities.

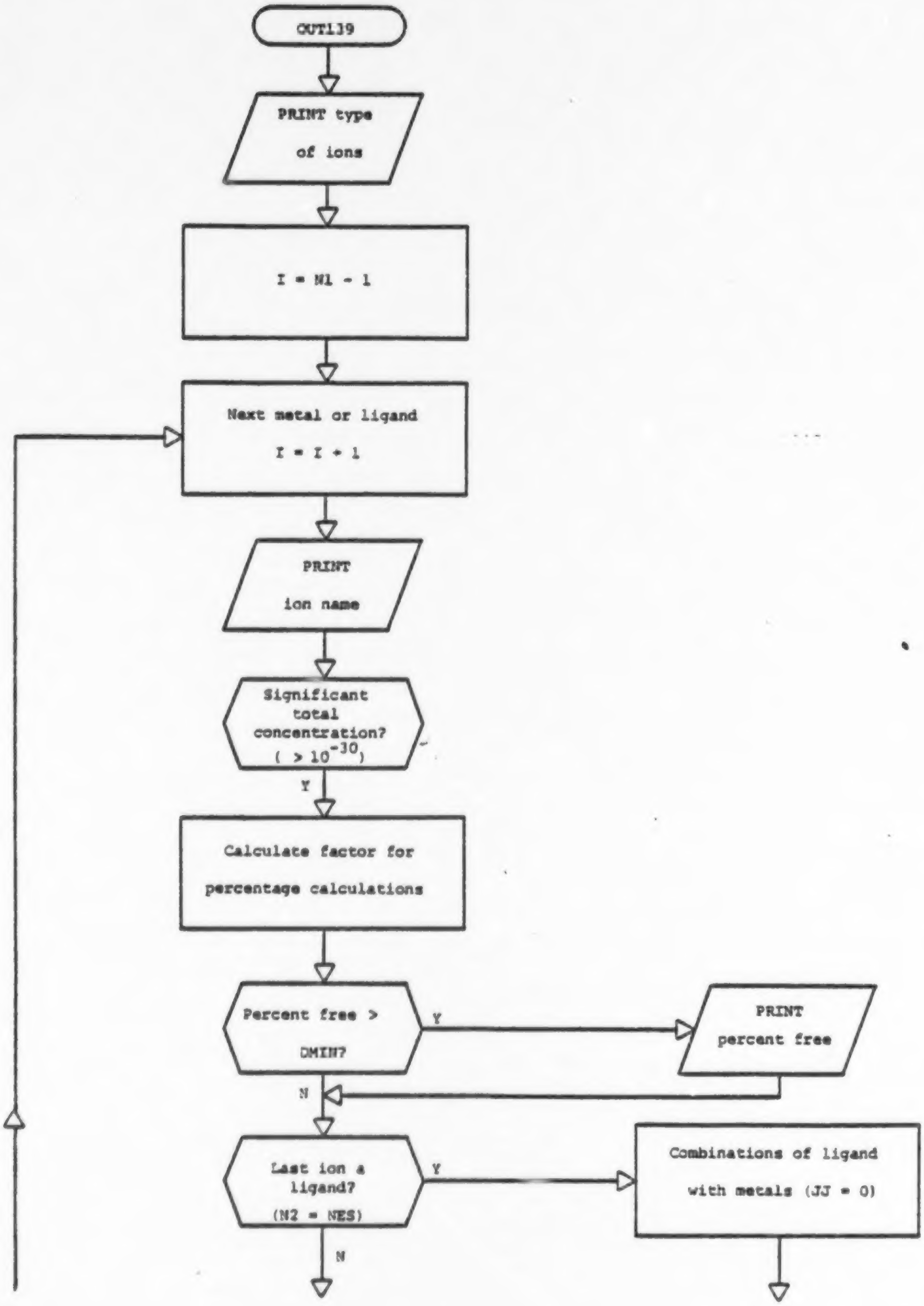

OUT17
PRINT heading for complexes
AFLAG = 0
Initalize for regular complexes LIMIT = NCREG, IEND = 0
LAST = IEND
Search for set CALL COLECT
Found one? (IEND = 0)
N
Y
Any absorption? (NC NCMIX, KADS = 0)
Y
N
Initialize for absorption KADS = 1, IEND = NCMIX, LIMIT = NC
Report activity? (AFLAG > 0)
Y
N
K = IBEG -1
K = IBEG -1
Next member of set K = K + 1

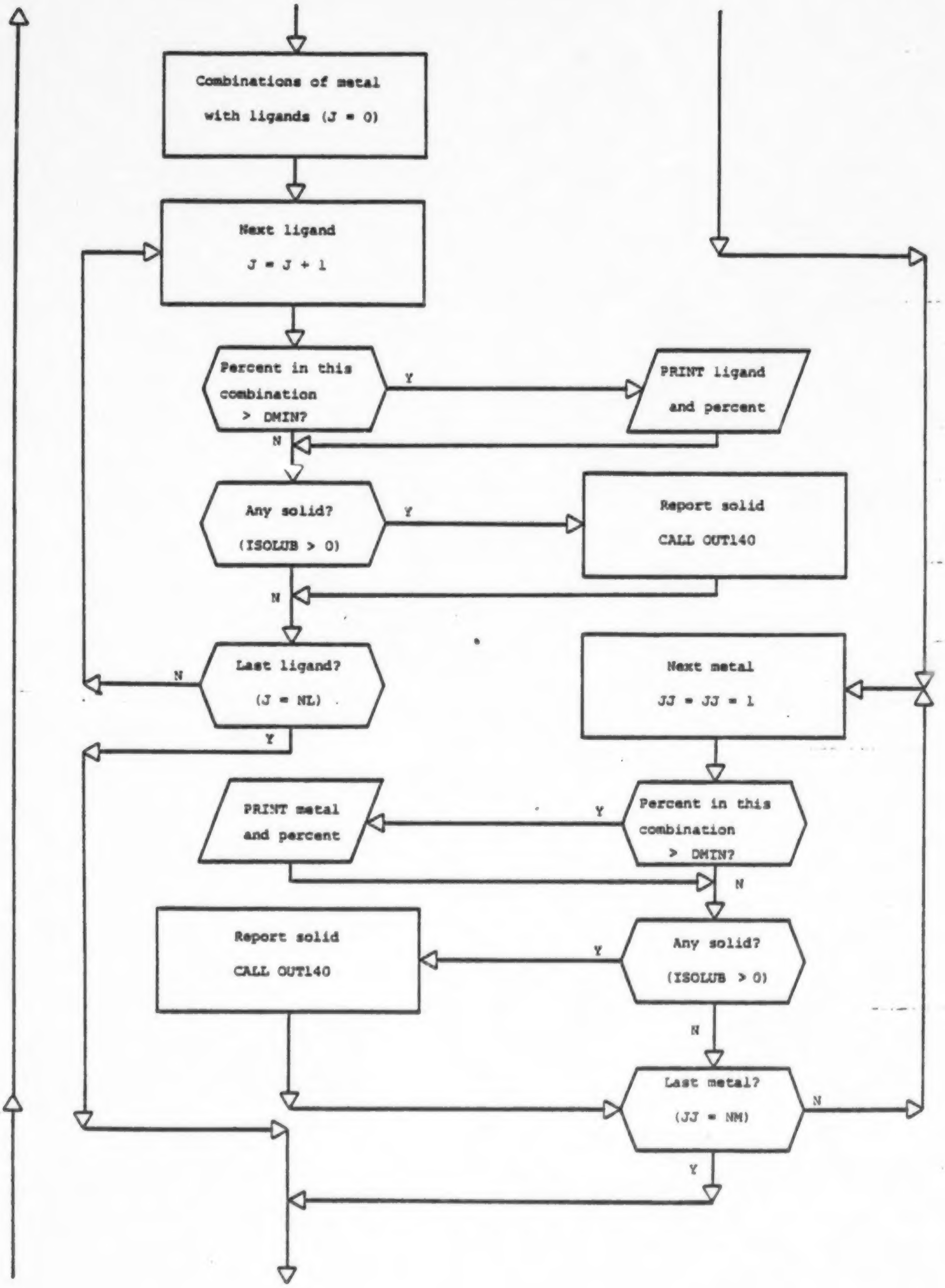

Next member of set
K = K + 1
Concentration too small? (< 10^-70)
Y
N
Concentration = 10^-100
Convert to -logM
K = IEND?
N
Y
Concentration = 10^-100
Concentration too small? (< 10^-70)
Y
Convert to -logM
Convert to activity ACTLOG
K = IEND?
Y
N
A new metal?
PRINT metal
PRINT ligand and complex concentrations and stoichiometries

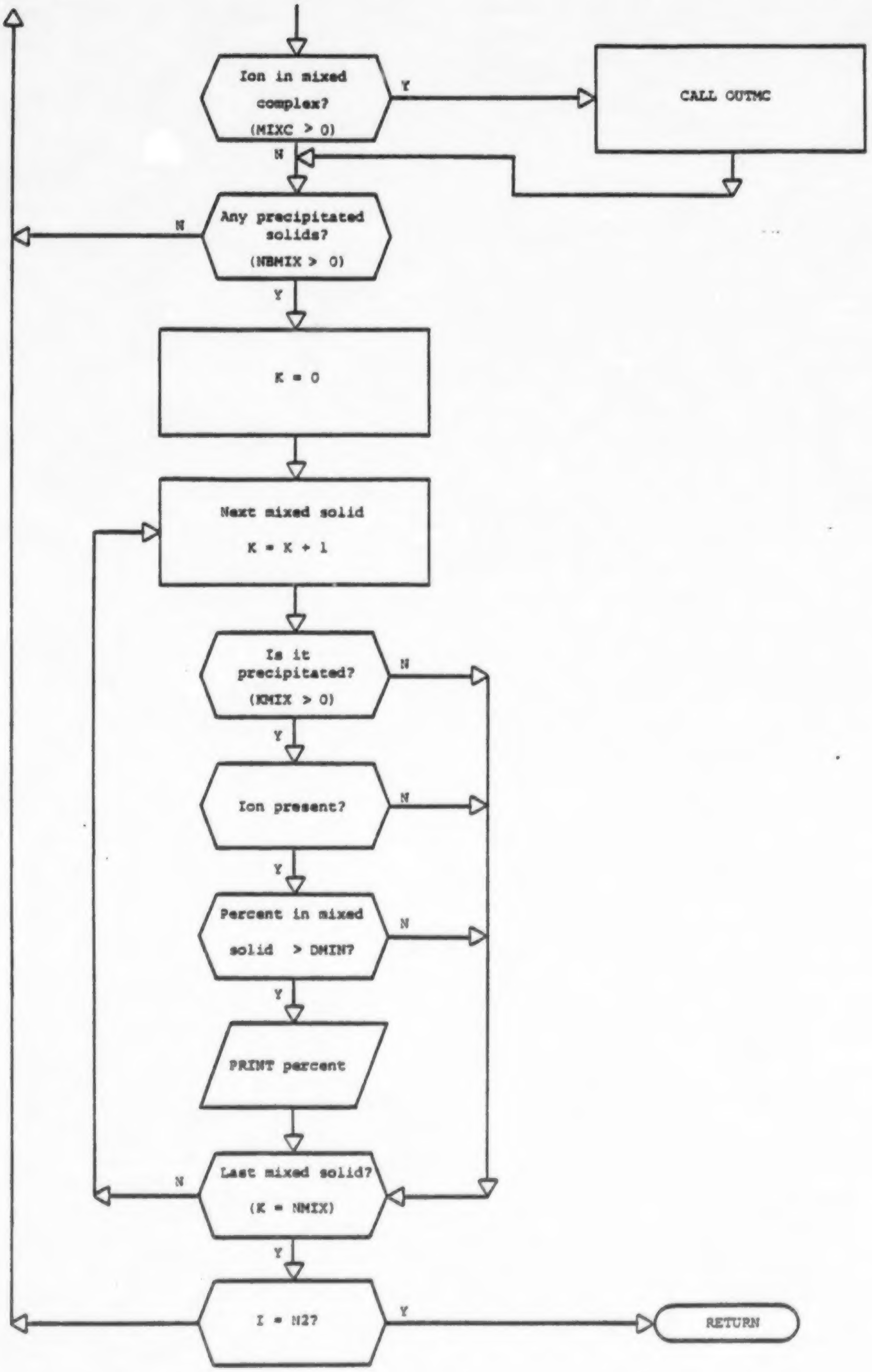

Any redox complexes?
(MCRED > 0)
N
Y
PRINT heading
K = 0
Next redox complex
K = K + 1
Calculate -logM
Activity?
(AFLAG = 1)
Y
N
Calculate -log activity
ACTLOG
PRINT components, stoichiometry and concentration
Last redox complex?
(K = MCRED)
N
Y
Mixed complexes?
(MCMIX > 0)
N
Y
PRINT heading
Report mixed complexes
CALL OUTMC
Report activities?
(IACT = 1, AFLAG = 0)
Y
N
RETURN
AFLAG = 1
PRINT heading

OUT139

OUT139 is called twice by SUB3 to report the "primary distributions" of metals and ligands; the amount of each metal involved in complexes and solids with each ligand is reported as a percentage of the total concentration of that metal. In the second call to OUT139, a similar report is made for ligands. Percentages which are less than an arbitrary value (DMIN) are not reported and any ions whose concentrations are less than 1.0E-30 are not reported. The majority of the routine is within a single loop which is executed once per ion. Amounts in complexes, solids, mixed complexes and mixed solids are reported in internal loops.

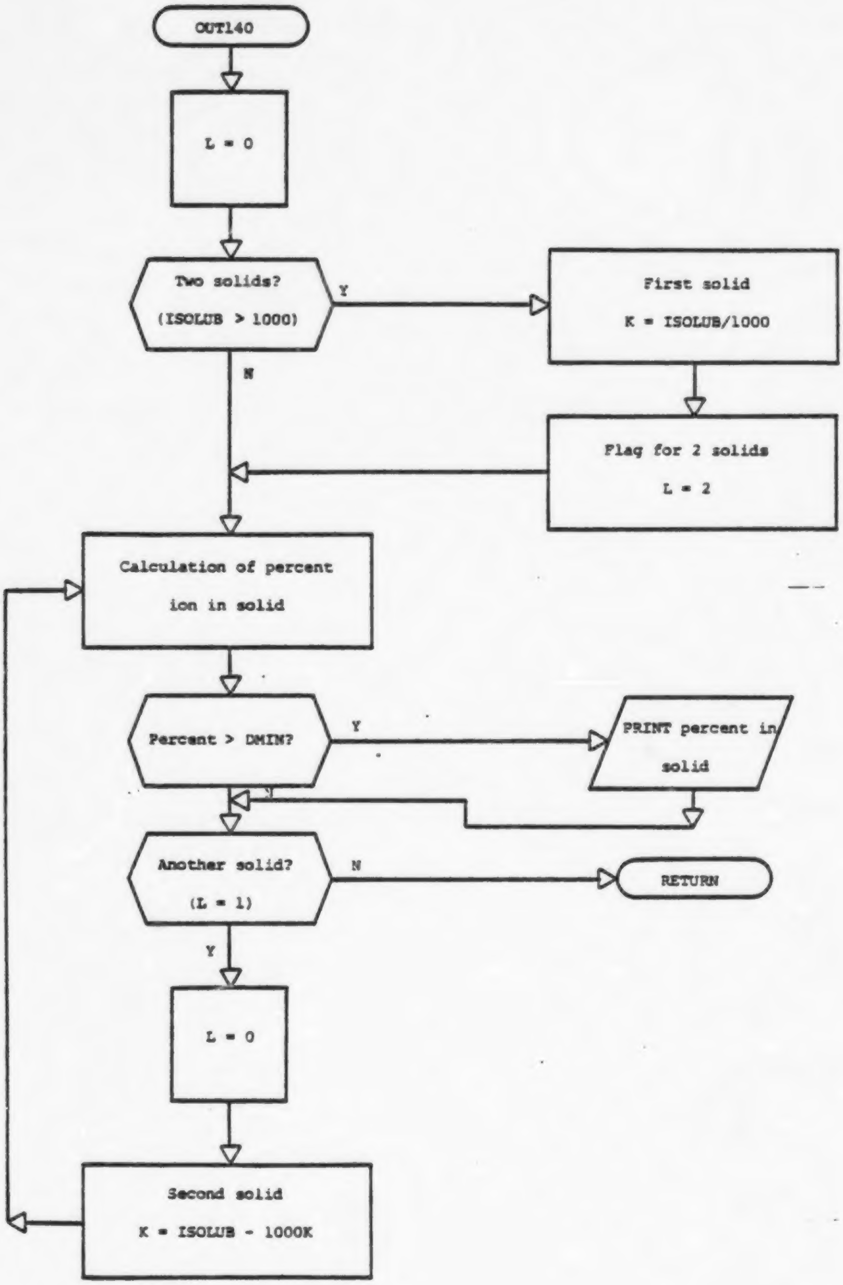

OUT139
PRINT type of ions
I = N1 - 1
Next metal or ligand
I = I + 1
PRINT ion name
Significant total concentration? (> 10^{-30})
Y
Calculate factor for percentage calculations
Percent free > DMIN?
Y
PRINT percent free
N
Last ion a ligand? (N2 = NES)
Y
Combinations of ligand with metals (JJ = 0)
N


Combinations of metal
with ligands (J = 0)
Next ligand
J = J + 1
Percent in this
combination
> DMIN?
Y
PRINT ligand
and percent
N
Any solid?
(ISOLUB > 0)
Y
Report solid
CALL OUT140
N
Last ligand?
(J = NL)
N
Y
Next metal
JJ = JJ = 1
Percent in this
combination
> DMIN?
Y
PRINT metal
and percent
N
Any solid?
(ISOLUB > 0)
Y
Report solid
CALL OUT140
N
Last metal?
(JJ = NM)
N
Y


Ion in mixed complex? (MIXC > 0)
Y
N
CALL OUTMC
Any precipitated solids? (NBMIX > 0)
N
Y
K = 0
Next mixed solid
K = K + 1
Is it precipitated? (KMIX > 0)
N
Y
Ion present?
N
Y
Percent in mixed solid > DMIN?
N
Y
PRINT percent
Last mixed solid? (K = NMIX)
N
Y
I = N2?
Y
RETURN

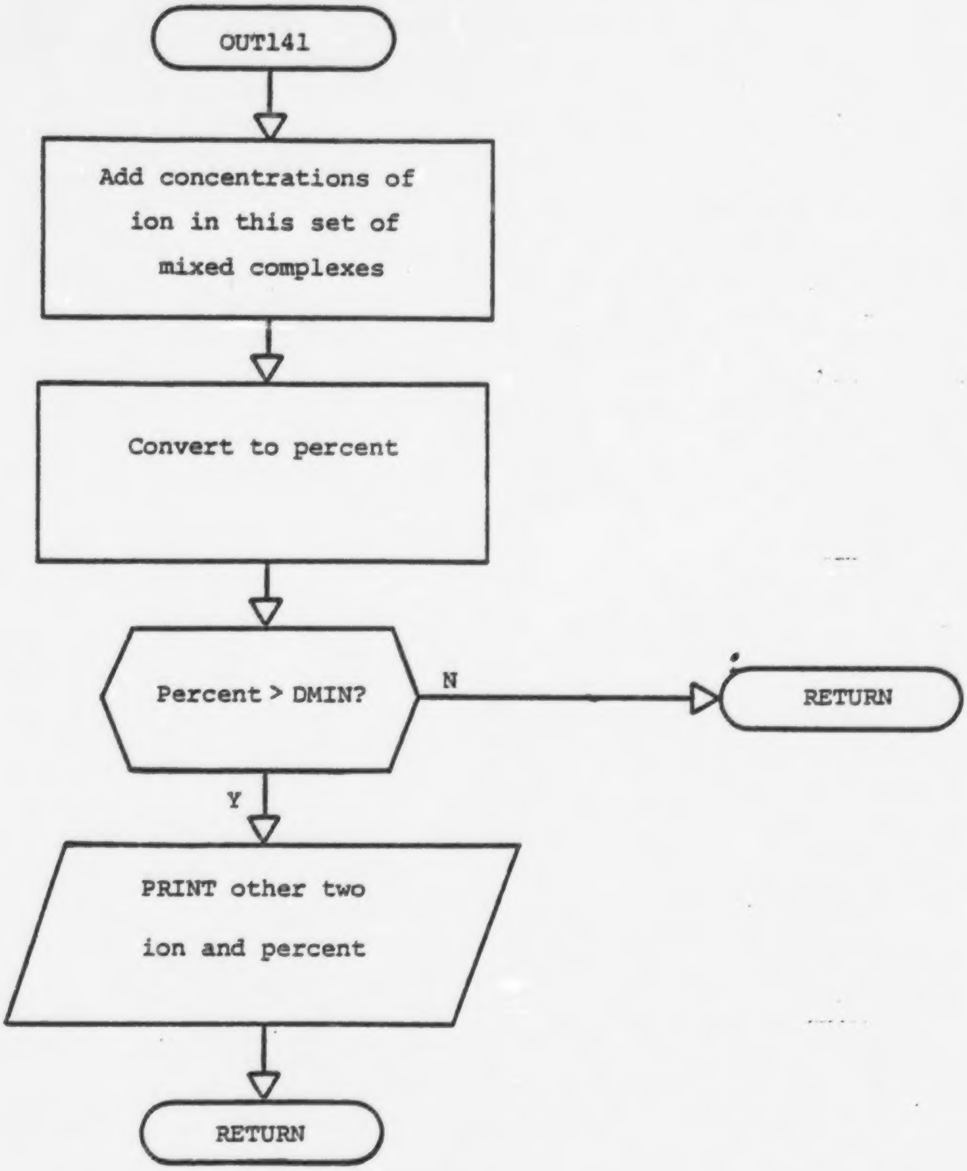
OUT140

OUT140 is called by OUT139 for a metal or ligand in a precipitated regular solid. OUT140 sorts out which solid(s) of the M-L pair is precipitated and reports the percentage of the ion in the solid.


OUT140
L = 0
Two solids?
(ISOLUB > 1000)
Y
First solid
K = ISOLUB/1000
N
Flag for 2 solids
L = 2
Calculation of percent
ion in solid
Percent > DMIN?
Y
PRINT percent in
solid
Another solid?
(L = 1)
N
RETURN
Y
L = 0
Second solid
K = ISOLUB - 1000K

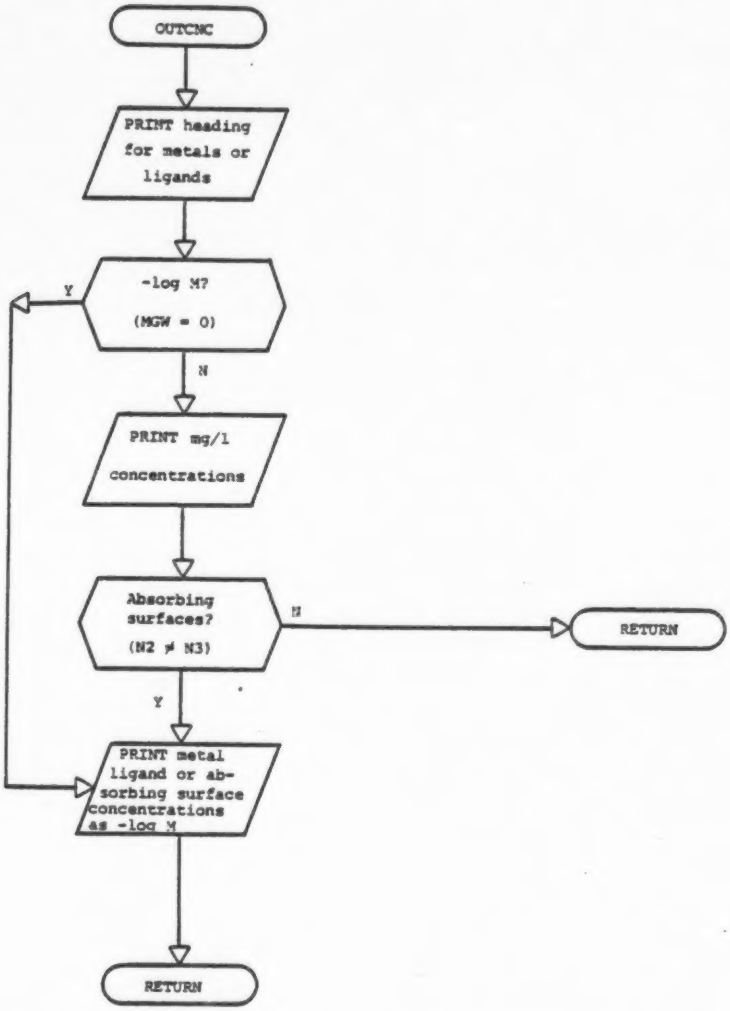
OUT141

OUT141 is used by OUTMC to compute and report the amount of an ion involved in mixed complexes with two other ions. OUT141 is called three times for each mixed complex, once for each ion in the complex.


OUT141
Add concentrations of ion in this set of mixed complexes
Convert to percent
Percent > DMIN?
N
RETURN
Y
PRINT other two ion and percent
RETURN

OUTCNC

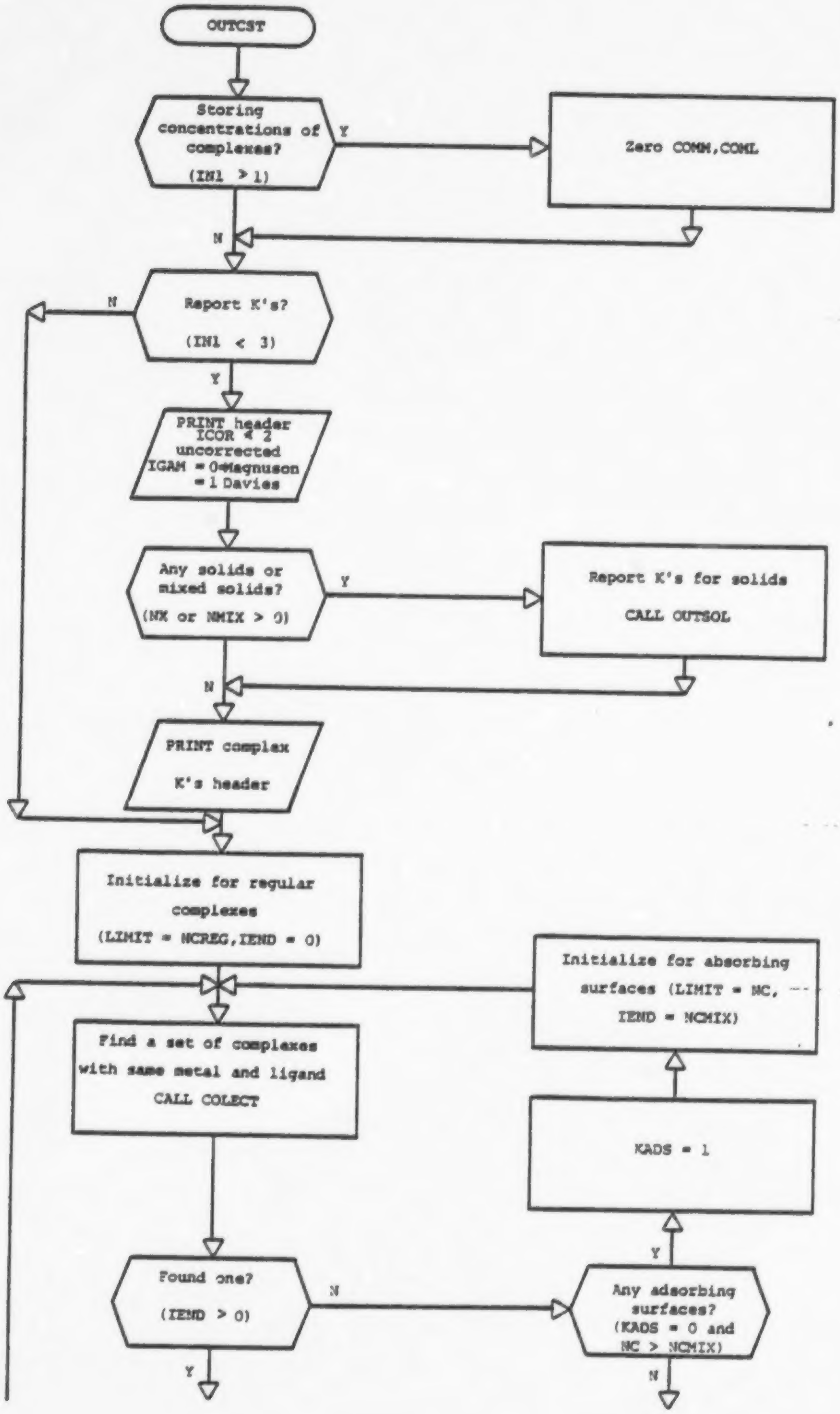
OUTCNC is used by OUTIN in echoing the user's input data. It reports total concentrations of each ion for each case. If concentrations were read as mg/liter, one loop is used; if they were read as molar concentrations, another loop is used. Any adsorbing surfaces are reported in molar units.

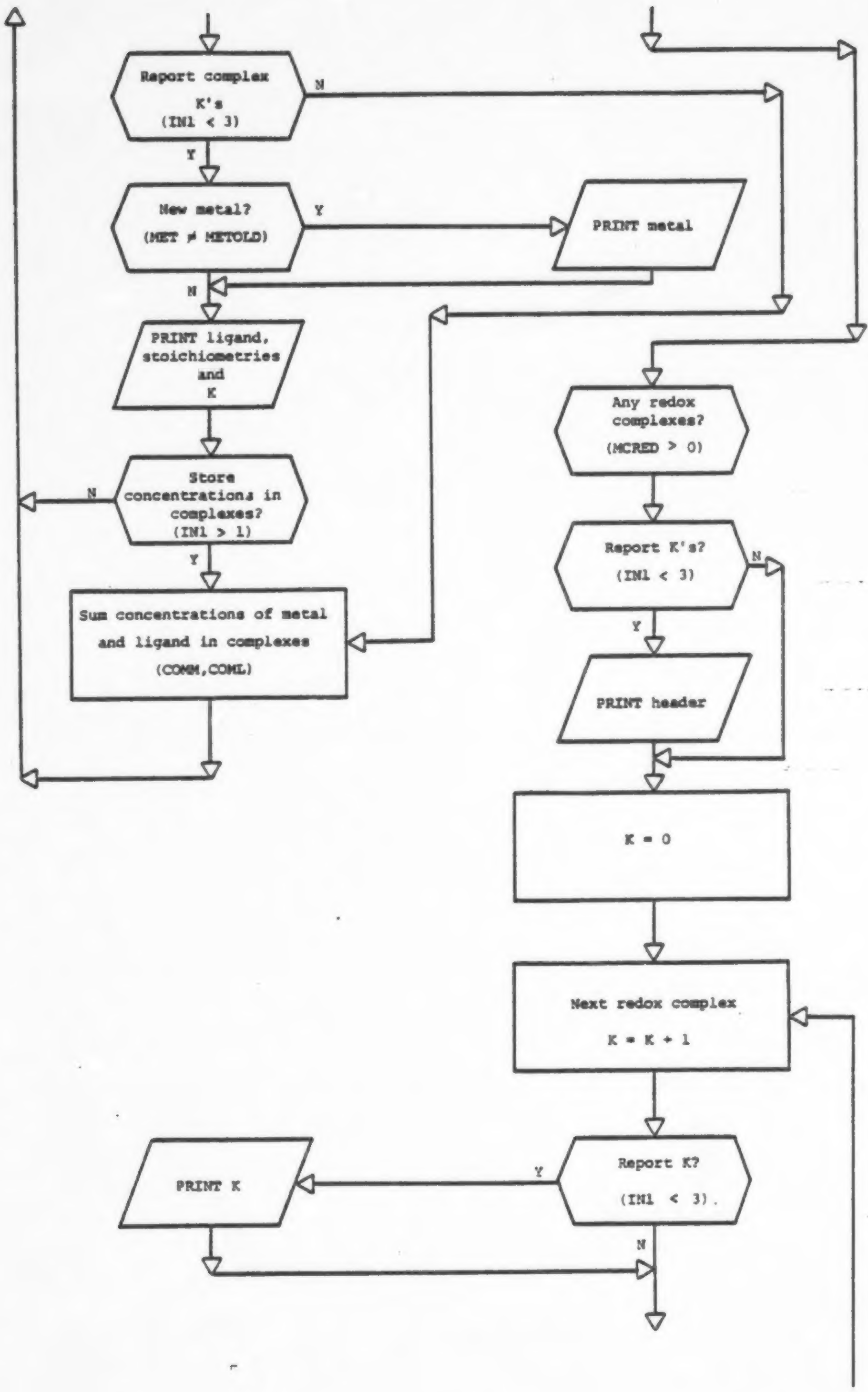

OUTCNC
PRINT heading for metals or ligands
-log M?
(MGW = 0)
Y
N
PRINT mg/l concentrations
Absorbing surfaces?
(N2 ≠ N3)
N
RETURN
Y
PRINT metal ligand or absorbing surface concentrations as -log M
RETURN

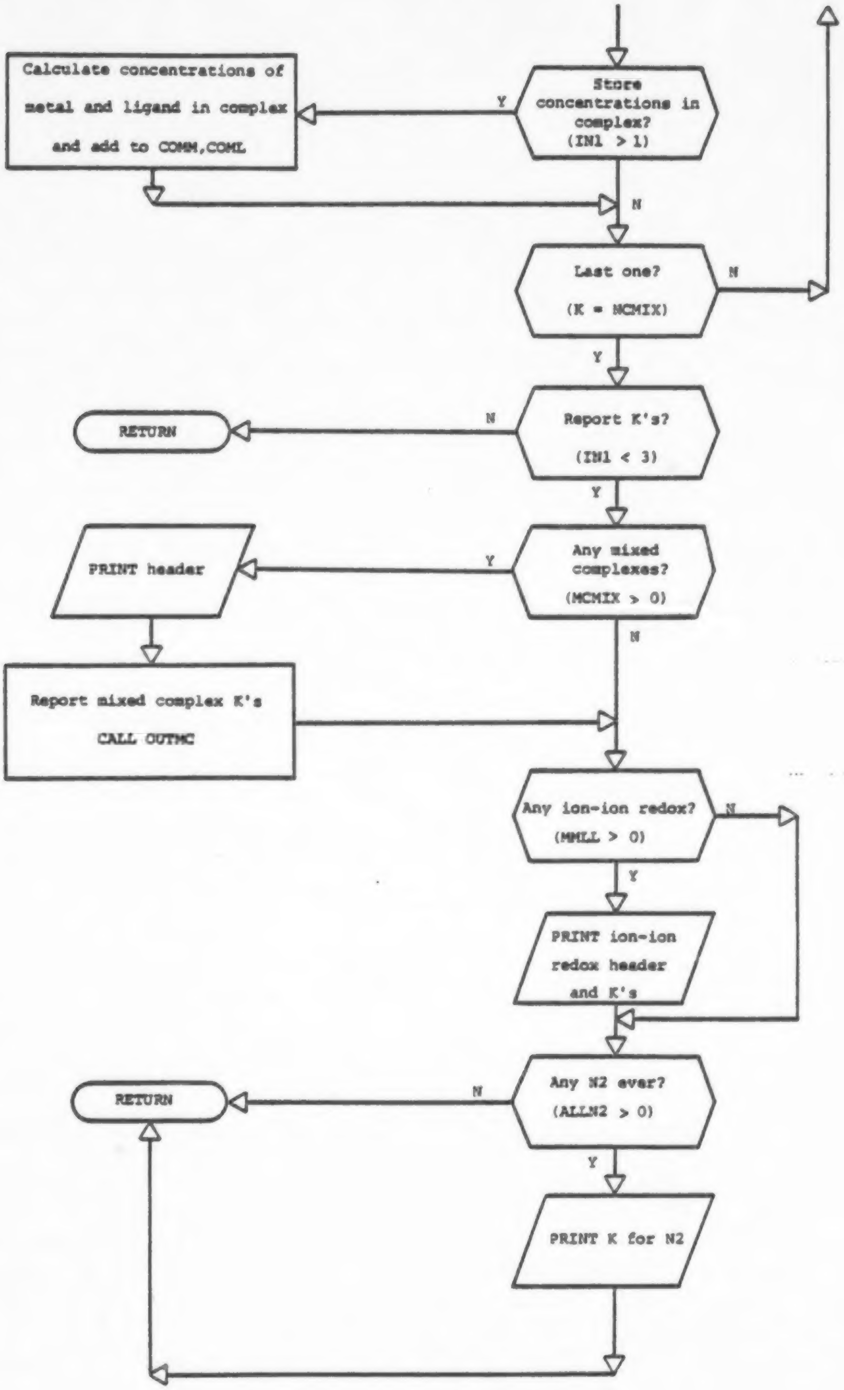
OUTCST

OUTCST is called by SUB2 in two different places and by SUB3 to report initial or final thermodynamic constants and to perform some initialization. The parameter IN1 determines the exact function to be performed; if IN1 = 1, thermodynamic data are reported. If IN1 = 2 thermodynamic data are reported and the concentrations of all complexes of each metal-ligand pair are summed for later use in OUT139. If IN1 = 3, output of thermodynamic data is suppressed but the summation of complexes is performed.

Nested loops are used to initialize the arrays COMM and COML which are used for summing complex concentrations. An upside-down loop reports thermodynamic data for possible solids (in conjunction with subroutine COLECT). The next two report possible redox and mixed solids respectively. The following upside-down loop reports thermodynamic data for complexes (again in conjunction with COLECT. Note that when IN1 = 1, a portion of the loop is not used). Thermodynamic data then are reported for any redox complexes, again performing summation of concentrations when IN1 = 2 or IN1 = 3. If any mixed complexes are in the system, OUTMC is used to report these. Ion-ion redox reactions, if any, then are reported. Finally, N2 gas is reported if present.

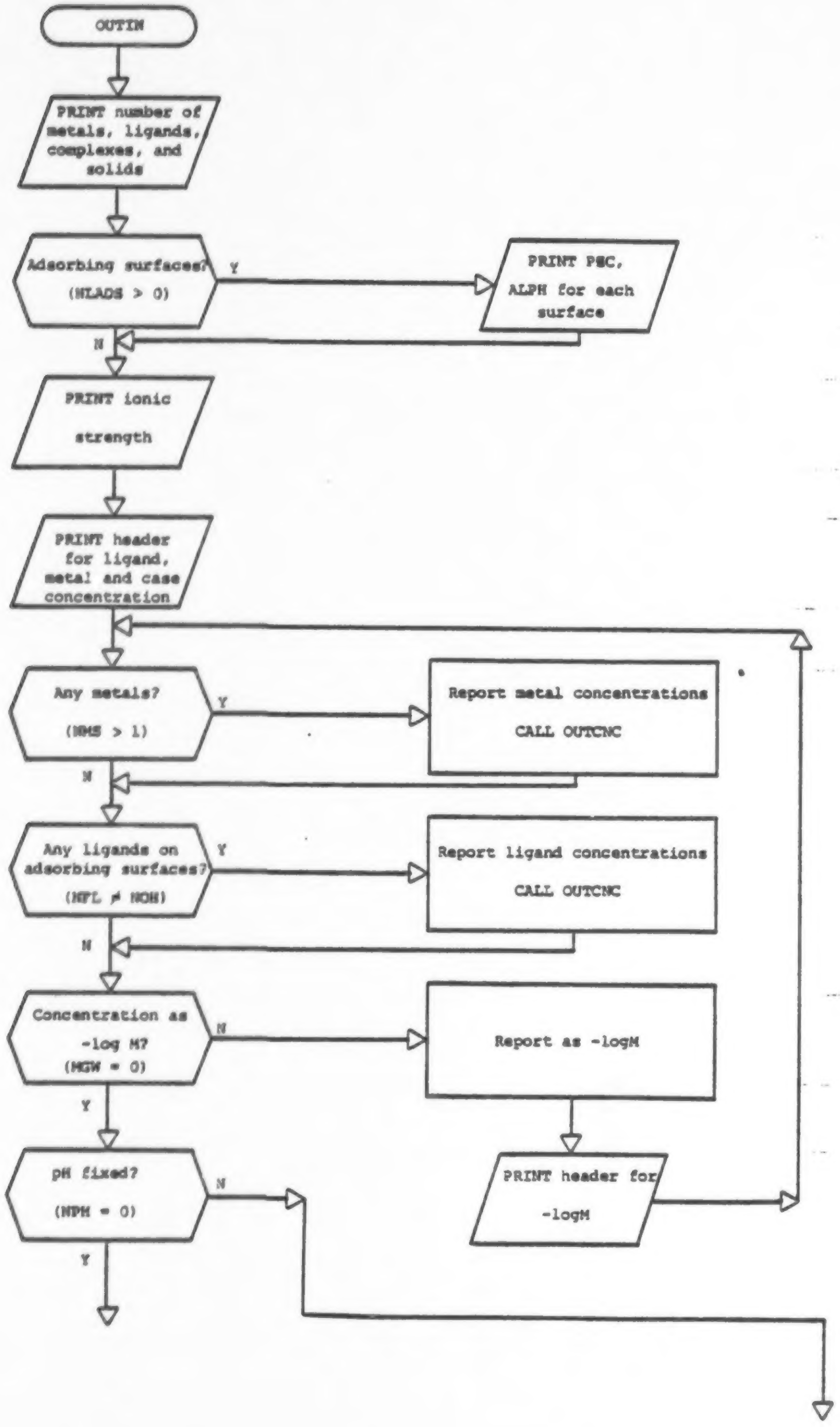

OUTCST
Storing concentrations of complexes? (IN1 > 1)
Y
Zero COMM,COML
N
Report K's? (IN1 < 3)
N
Y
PRINT header ICOR < 2 uncorrected IGAM = 0 Magnuson = 1 Davies
Any solids or mixed solids? (NX or NMIX > 0)
Y
Report K's for solids CALL OUTSOL
N
PRINT complex K's header
Initialize for regular complexes (LIMIT = NCREG, IEND = 0)
Initialize for absorbing surfaces (LIMIT = NC, IEND = NCMIX)
Find a set of complexes with same metal and ligand CALL COLECT
KADS = 1
Y
Found one? (IEND > 0)
N
Any adsorbing surfaces? (KADS = 0 and NC > NCMIX)
Y
N

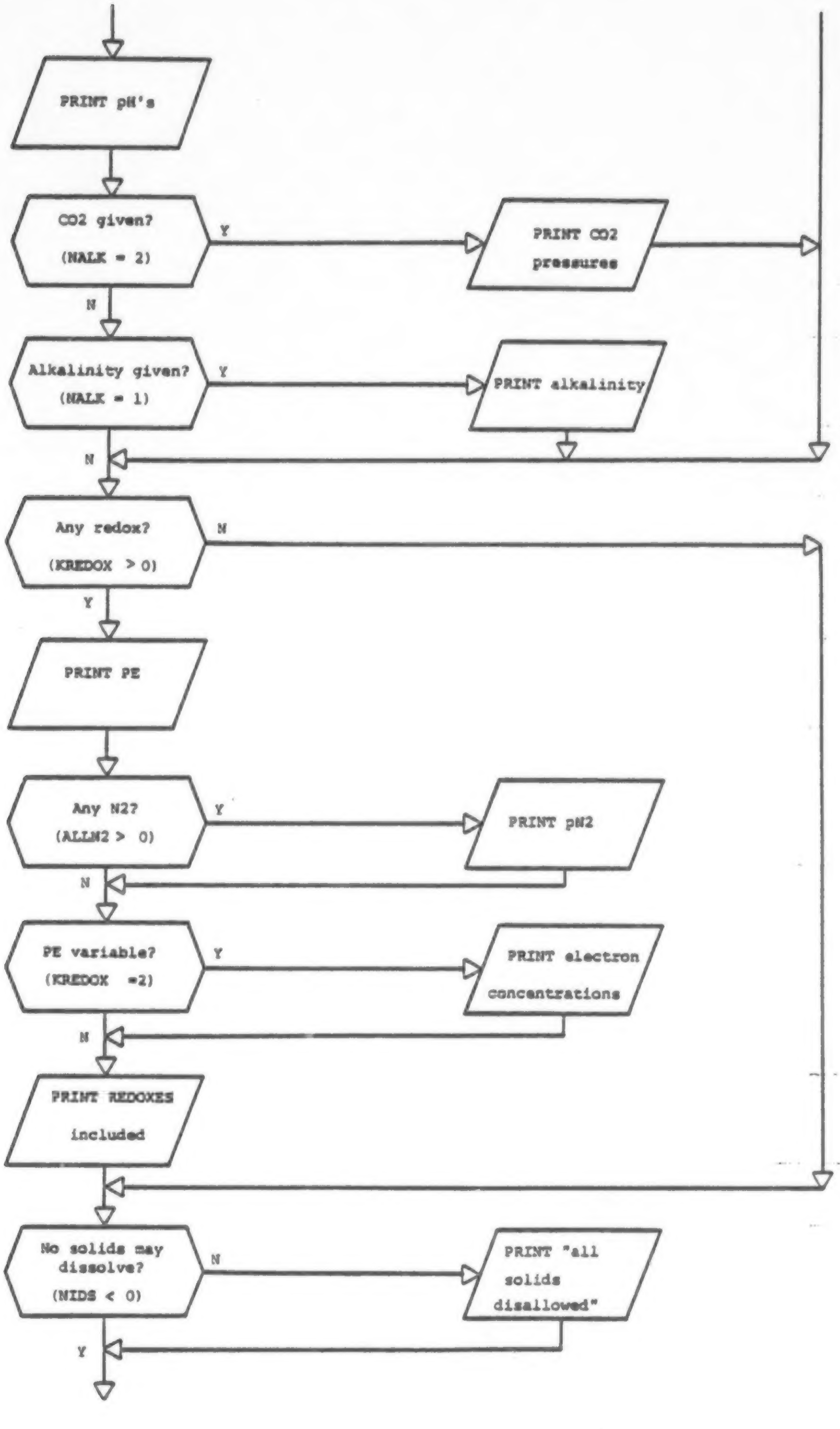

Report complex K's (IN1 < 3)
N
Y
New metal? (MET ≠ METOLD)
Y
PRINT metal
N
PRINT ligand, stoichiometries and K
Store concentrations in complexes? (IN1 > 1)
N
Y
Sum concentrations of metal and ligand in complexes (COMM,COML)
Any redox complexes? (MCRED > 0)
Report K's? (IN1 < 3)
N
Y
PRINT header
K = 0
Next redox complex
K = K + 1
Report K? (IN1 < 3)
Y
PRINT K
N

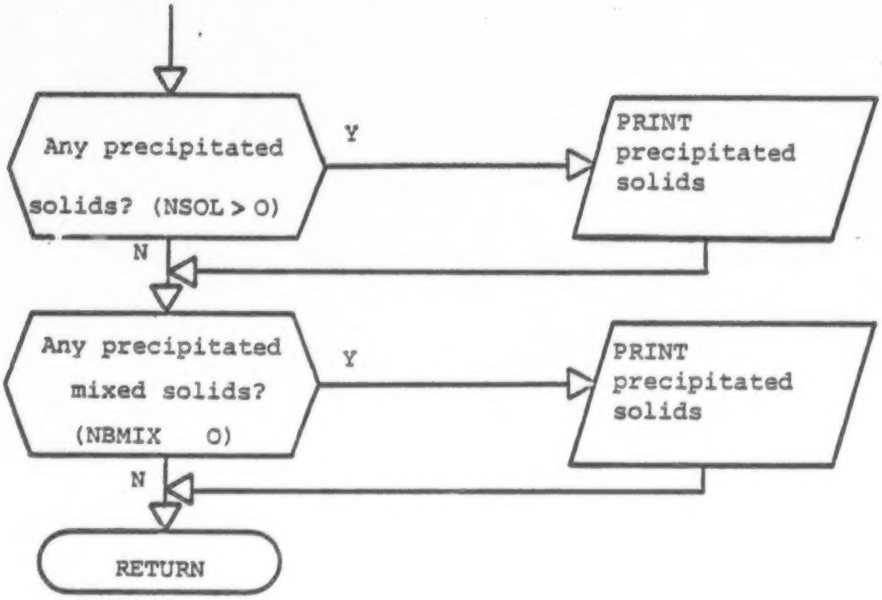

Calculate concentrations of metal and ligand in complex and add to COMM,COML
Store concentrations in complex? (IN1 > 1)
Y
N
Last one? (K = NCMIX)
N
Y
RETURN
N
Report K's? (IN1 < 3)
Y
PRINT header
Y
Any mixed complexes? (MCMIX > 0)
N
Report mixed complex K's CALL OUTMC
Any ion-ion redox? (MMLL > 0)
N
Y
PRINT ion-ion redox header and K's
RETURN
N
Any N2 ever? (ALLN2 > 0)
Y
PRINT K for N2

OUTIN

Subroutine OUTIN is called by SUB2 to echo part of the user's input data. Initially, the number of metals, ligands, complexes, and possible solids are reported. This is followed (in order) by adsorption data (if any), ionic strength, conditions for each case (in conjunction with OUTCNC), pH, CO2 partial pressures (if any), redox data (if any), imposed solids (if any), and finally imposed mixed solids (if any).

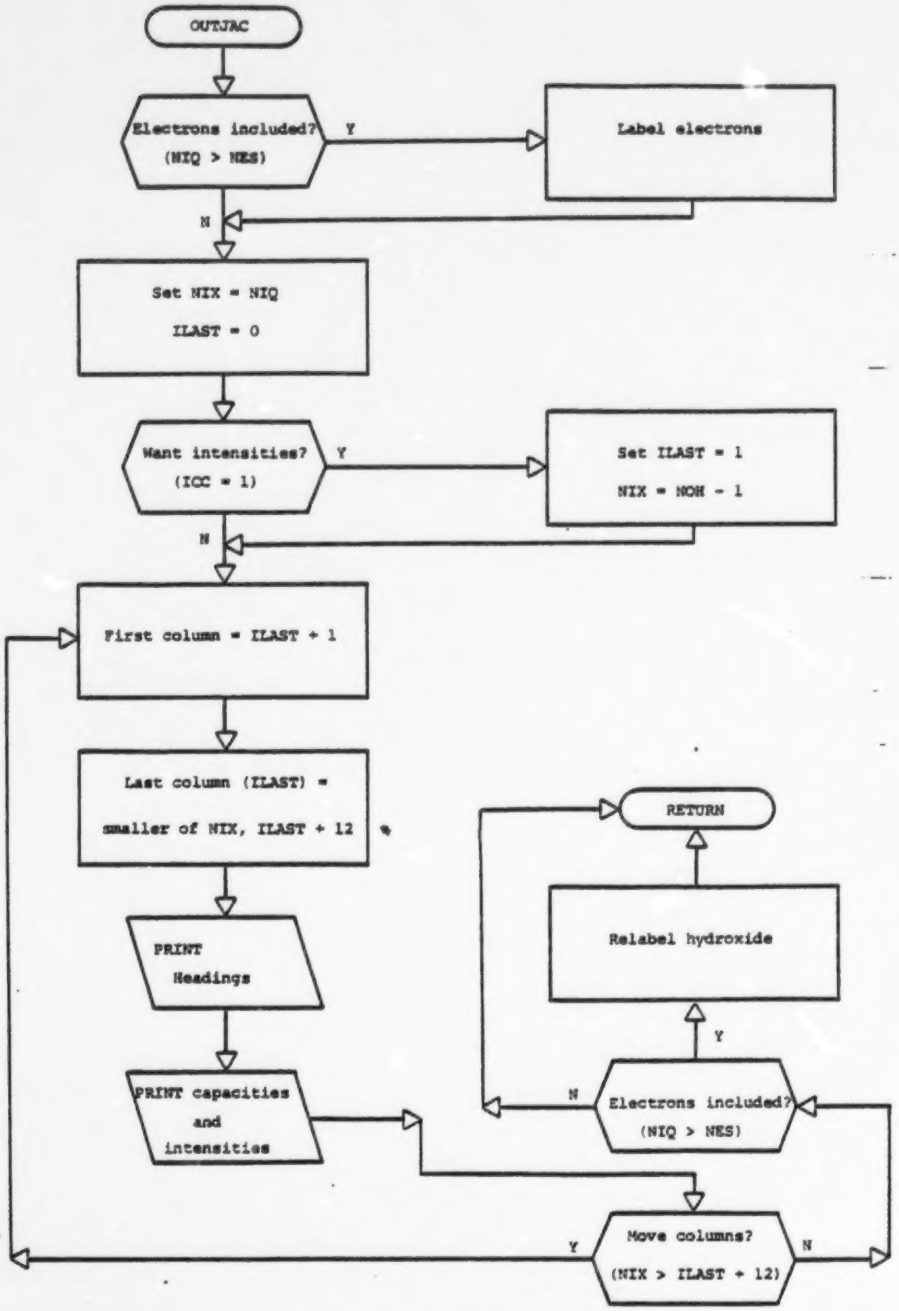

OUTIN
PRINT number of metals, ligands, complexes, and solids
Adsorbing surfaces? (NLADS > 0)
Y
N
PRINT PSC, ALPH for each surface
PRINT ionic strength
PRINT header for ligand, metal and case concentration
Any metals? (NMS > 1)
Y
N
Report metal concentrations CALL OUTCNC
Any ligands on adsorbing surfaces? (NFL ≠ NOH)
Y
N
Report ligand concentrations CALL OUTCNC
Concentration as -log M? (MGW = 0)
N
Y
Report as -logM
PRINT header for -logM
pH fixed? (NPH = 0)
N
Y


PRINT pH's
CO2 given?
(NALK = 2)
Y
PRINT CO2
pressures
N
Alkalinity given?
(NALK = 1)
Y
PRINT alkalinity
N
Any redox?
(KREDOX > 0)
N
Y
PRINT PE
Any N2?
(ALLN2 > 0)
Y
PRINT pN2
N
PE variable?
(KREDOX =2)
Y
PRINT electron
concentrations
N
PRINT REDOXES
included
No solids may
dissolve?
(NIDS < 0)
N
PRINT "all
solids
disallowed"
Y


Any precipitated solids? (NSOL > O)
Y
PRINT precipitated solids
N
Any precipitated mixed solids? (NBMIX O)
Y
PRINT precipitated solids
N
RETURN

OUTJAC

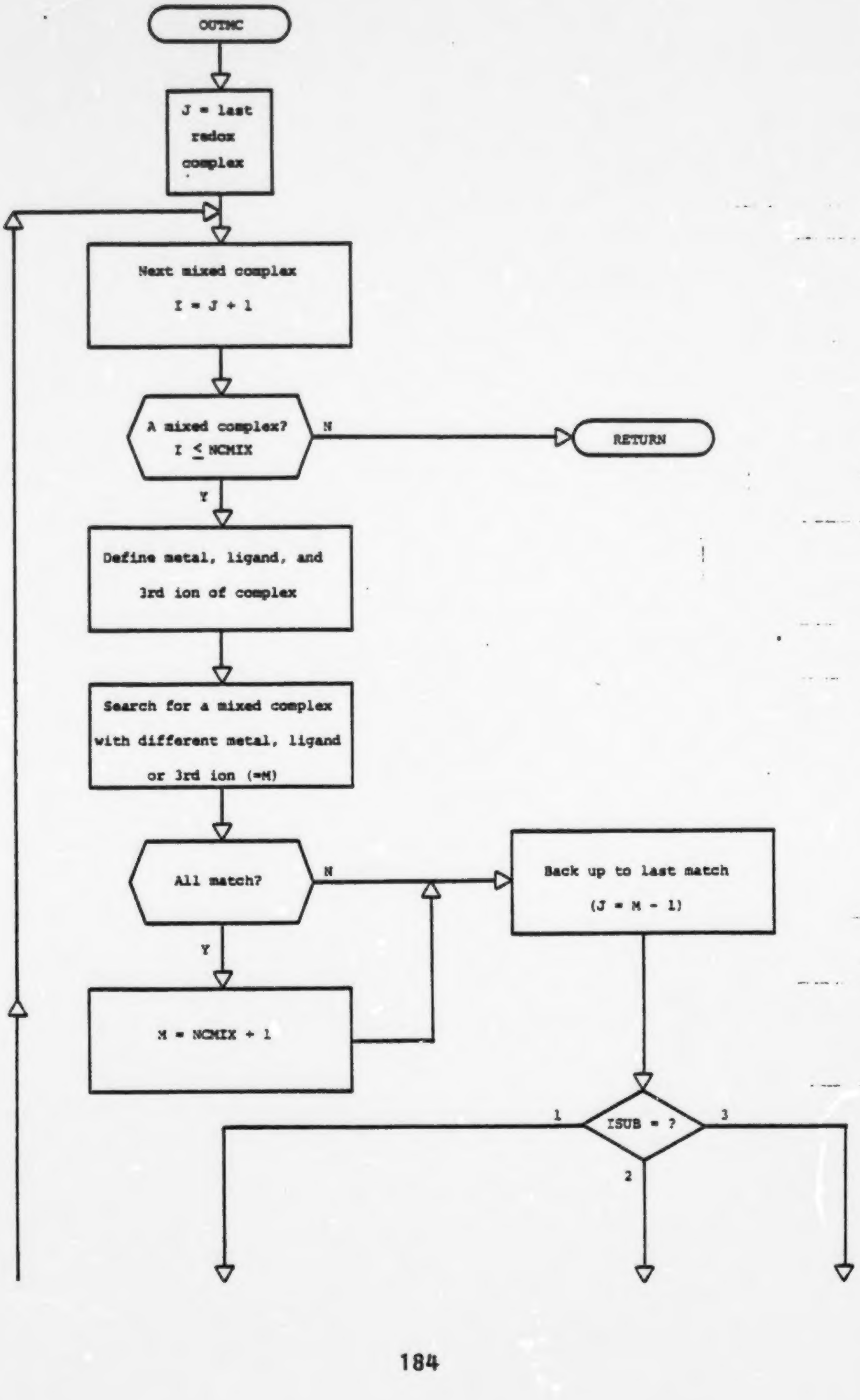
OUTJAC is called by SUBJAC to report interaction capacities (ICC=0) for metals, ligands, hydrogen and electrons, or intensities (ICC=1) for metals and ligands.

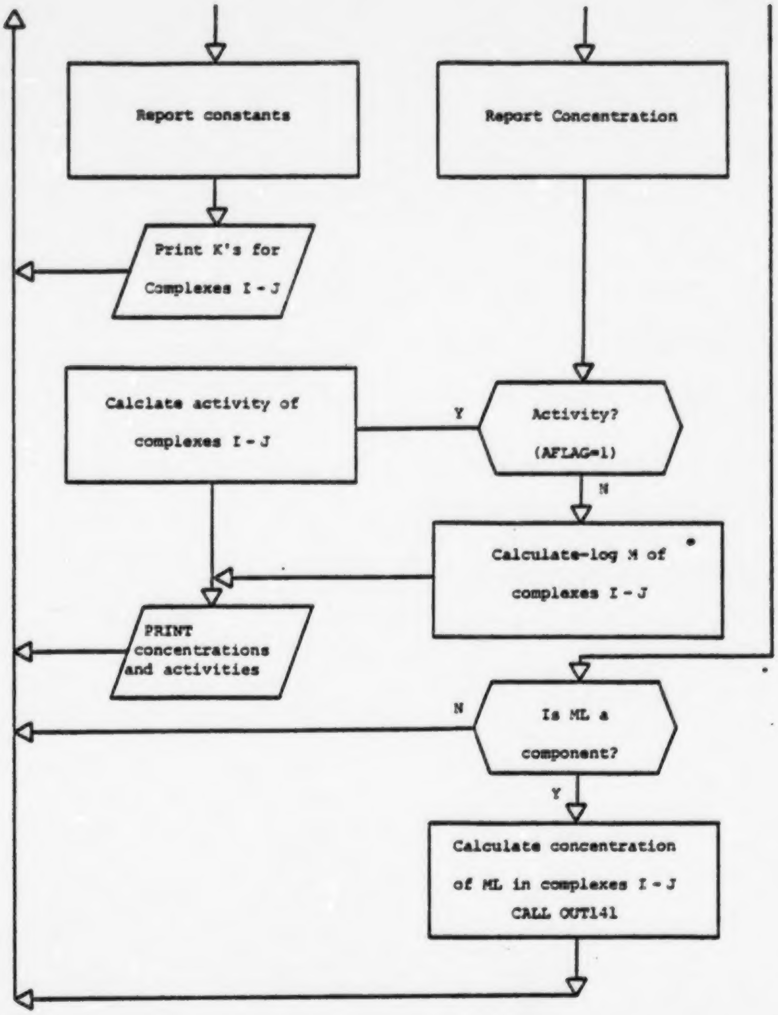

OUTJAC
Electrons included?
(NIQ > NES)
Y
Label electrons
N
Set NIX = NIQ
ILAST = 0
Want intensities?
(ICC = 1)
Y
Set ILAST = 1
NIX = NOH - 1
N
First column = ILAST + 1
Last column (ILAST) =
smaller of NIX, ILAST + 12
PRINT
Headings
PRINT capacities
and
intensities
RETURN
Relabel hydroxide
Y
N
Electrons included?
(NIQ > NES)
Y
Move columns?
(NIX > ILAST + 12)
N

OUTMC

OUTMC is used by OUTCST, OUT17, and OUT139 to report mixed complex information. OUTMC reports thermodynamic data when called by OUTCST (ISUB=1), concentrations of complexes when called by OUT17 (ISUB=2), and percentage concentration of the ions in mixed complexes (as done for "regular" complexes by OUT139) when called by OUT139 (ISUB=3). OUTMC is called with ML, the ion whose percentages are being computed, when called by OUT139, and with AFLAG when called by OUT17.

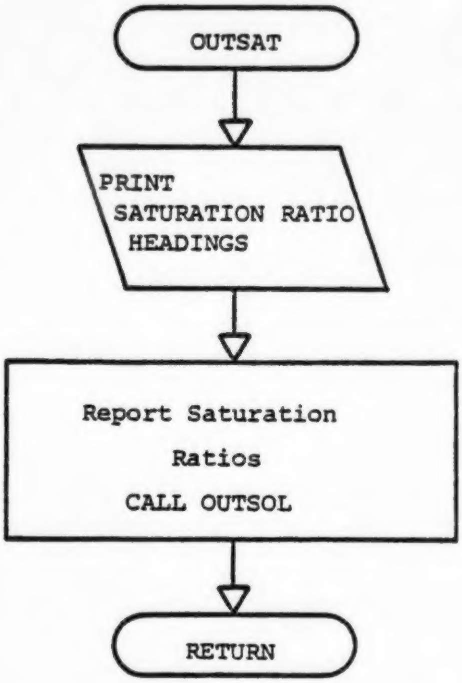
The first several lines of the routine perform initialization and find the position of the last mixed complex which has the same components as mixed complex "I". If I is greater than NCMIX, either there are no mixed complexes or all data has been reported; in either case a RETURN is executed. Otherwise, a computed GOTO selects the form of output to be used, based on the value of ISUB. After printing, the pointer I is incremented to the last reported plus one and the next set of mixed complexes are processed. Note that the value of AFLAG controls whether data are reported as -log M or -log activities when OUTMC is called by OUT17.


OUTMC
J = last redox complex
Next mixed complex
I = J + 1
A mixed complex?
I ≤ NCMIX
N
RETURN
Y
Define metal, ligand, and 3rd ion of complex
Search for a mixed complex with different metal, ligand or 3rd ion (=M)
All match?
N
Back up to last match
(J = M - 1)
Y
M = NCMIX + 1
ISUB = ?
1
2
3


Report constants
Report Concentration
Print K's for
Complexes I - J
Calclate activity of
complexes I - J
Y
Activity?
(AFLAG=1)
N
Calculate-log M of
complexes I - J
PRINT
concentrations
and activities
N
Is ML a
component?
Y
Calculate concentration
of ML in complexes I - J
CALL OUT141

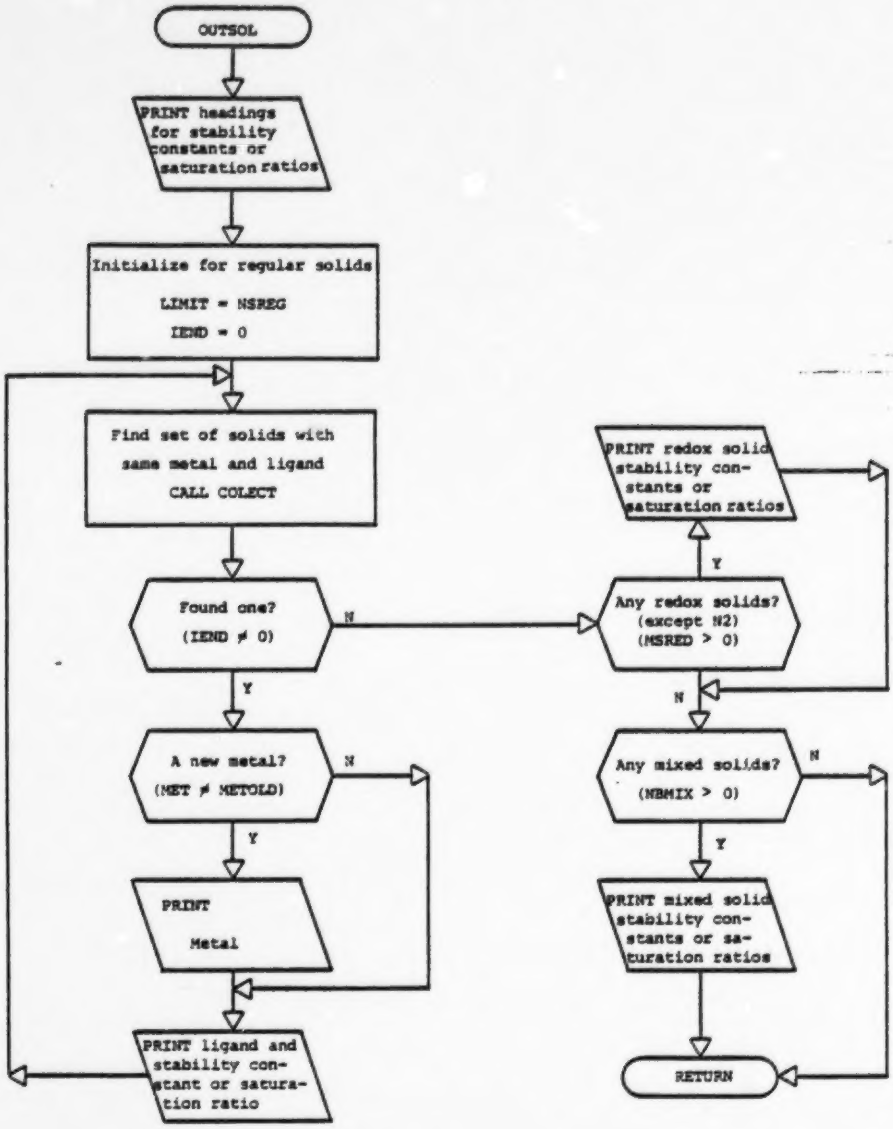
OUTSAT

OUTSAT is used by SUB3 to report the degree of saturation of each solid. It calls OUTSOL for printing degree of saturation.


OUTSAT
PRINT
SATURATION RATIO
HEADINGS
Report Saturation
Ratios
CALL OUTSOL
RETURN

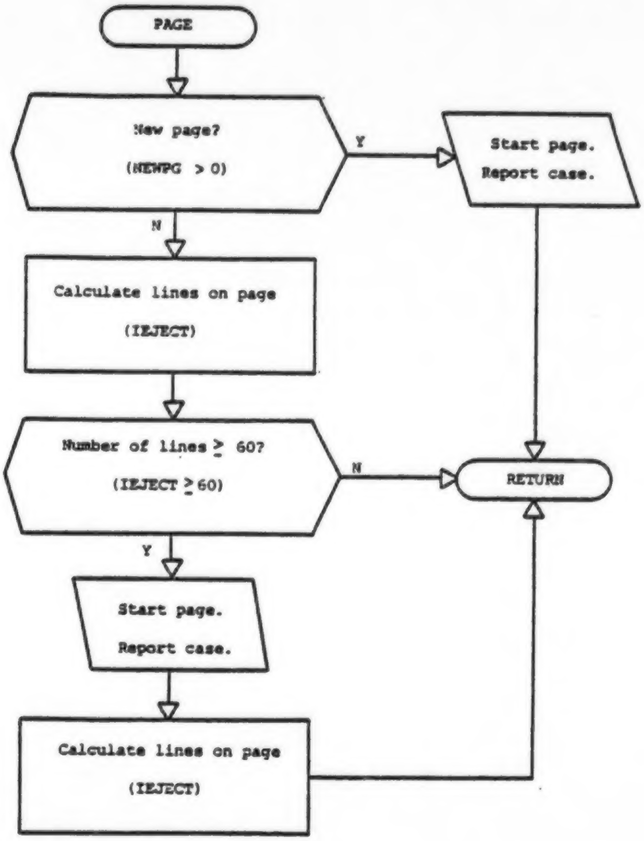
OUTSOL

OUTSOL prints output data for solids. Stability constants or saturation ratios are reported for regular, redox, and mixed solids.


OUTSOL
PRINT headings for stability constants or saturation ratios
Initialize for regular solids
LIMIT = NSREG
IEND = 0
Find set of solids with same metal and ligand
CALL COLECT
Found one?
(IEND ≠ 0)
N
Y
A new metal?
(MET ≠ METOLD)
N
Y
PRINT
Metal
PRINT ligand and stability constant or saturation ratio
PRINT redox solid stability constants or saturation ratios
Y
Any redox solids? (except N2)
(NSRED > 0)
N
Any mixed solids?
(NSMIX > 0)
N
Y
PRINT mixed solid stability constants or saturation ratios
RETURN

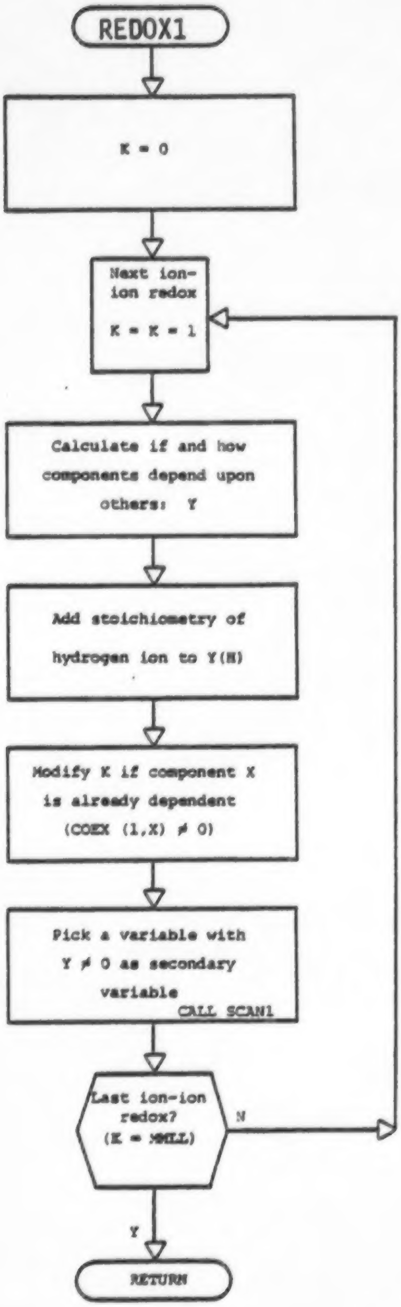
PAGE

PAGE is an uncomplicated routine which counts the number of lines printed and performs page ejects as necessary in order to leave at least three blank lines at the top of each page and four at the bottom. It is called with two variables: LINES, the number of lines about to print, and NEWPG, which is one if a new output routine is beginning to print. It is used by all output routines.


PAGE
New page?
(NEWPG > 0)
Y
Start page.
Report case.
N
Calculate lines on page
(IEJECT)
Number of lines ≥ 60?
(IEJECT ≥ 60)
N
RETURN
Y
Start page.
Report case.
Calculate lines on page
(IEJECT)

REDOX1

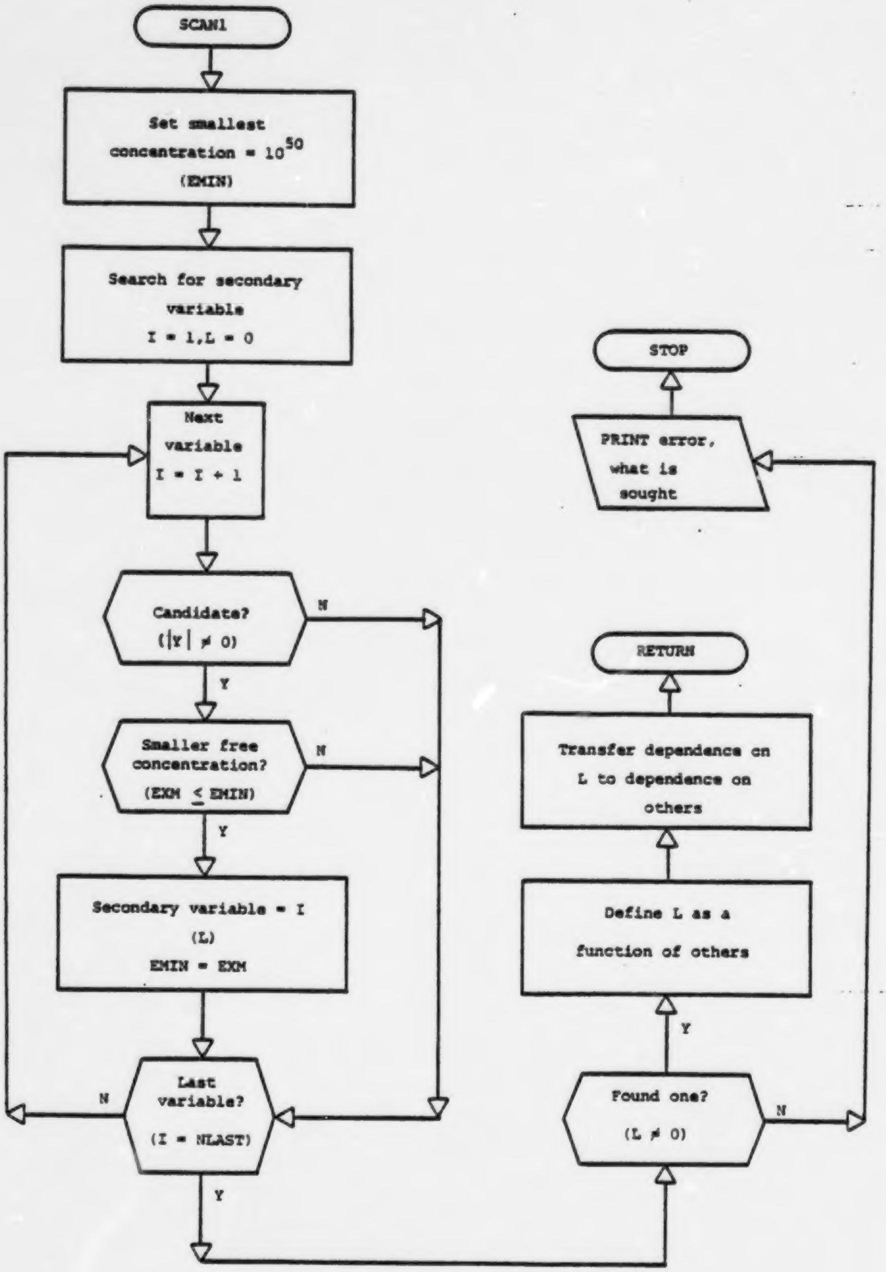
REDOX1 is used by SUB3 to select secondary variables due to ion-ion redox reactions. The reactions are treated one at a time. A list of candidates for the secondary variable is selected by computing Y, which will be nonzero for variables upon which the ions or electrons depend. The stability constant for the reaction is modified to reflect the effects of pre-existing secondary variables. Finally, SCAN1 is called to complete the definition of the new secondary variable and the process is begun for the next reaction.


REDOX1
K = 0
Next ion-
ion redox
K = K = 1
Calculate if and how
components depend upon
others: Y
Add stoichiometry of
hydrogen ion to Y(H)
Modify K if component X
is already dependent
(COEX (1,X) ≠ 0)
Pick a variable with
Y ≠ 0 as secondary
variable
CALL SCAN1
Last ion-ion
redox?
(K = MMLL)
N
Y
RETURN

SCAN1

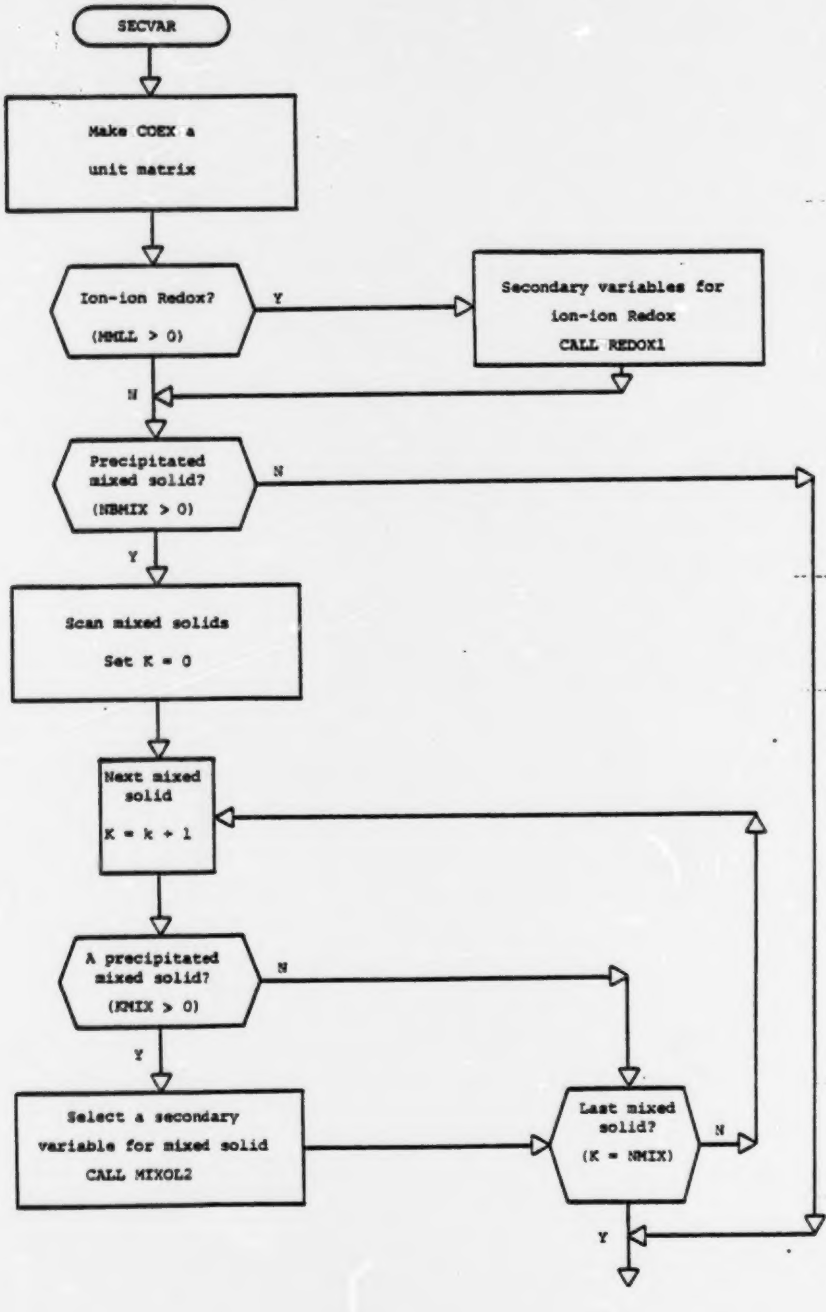
SCAN1 is used by REDOX1, MIXOL2, and SOLID2 to complete the task of defining a new secondary variable. A list of candidates for the new secondary variable has been selected by the calling routine prior to the call to SCAN1; these have nonzero values in the Y array. Upon entry to SCAN1, the three parameters CX, INDX, and ITYP are the (modified) negative log formation constant for the reaction creating the new secondary variable, the position of the data for the reaction (among solids, mixed solids, etc.) and the type (i.e. 1 = "normal" solid, 2 = mixed solid, and 3 = ion-ion redox) of the reaction.

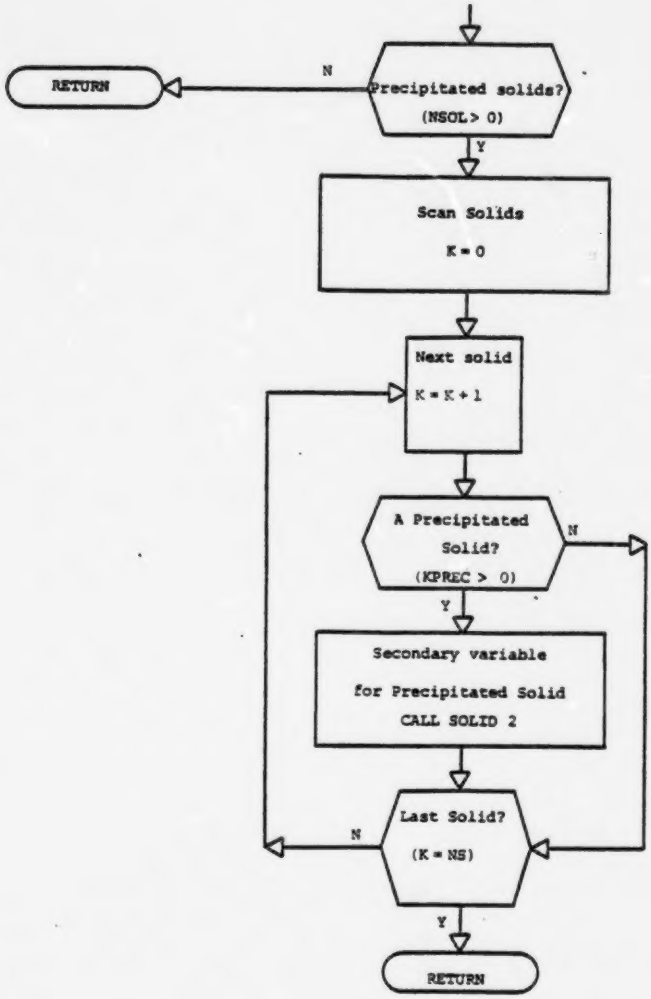
The concentration of the ion selected as the new secondary variable will hereafter be defined in terms of the concentrations of the other ion(s) in the solid or redox. If no secondary variable is found, a violation of the Gibbs Phase Rule has occurred, an informative message is printed, and the run is aborted.


SCAN1
Set smallest concentration = 10^50 (EMIN)
Search for secondary variable I = 1,L = 0
Next variable I = I + 1
Candidate? (|Y| ≠ 0)
N
Y
Smaller free concentration? (EXM ≤ EMIN)
N
Y
Secondary variable = I (L) EMIN = EXM
Last variable? (I = NLAST)
N
Y
Found one? (L ≠ 0)
N
Y
Define L as a function of others
Transfer dependence on L to dependence on others
RETURN
PRINT error, what is sought
STOP

SECVAR

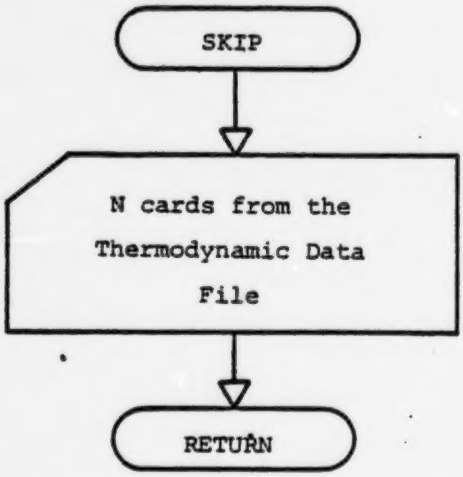
SECVAR is used by SUB3 to compute the COEX matrix, defining a secondary variable for each solid, and ion-ion redox reaction. The routine begins with a unit matrix indicating that each variable is dependent only upon itself. The matrix then is modified for each secondary variable found.


SECVAR
Make COEX a unit matrix
Ion-ion Redox? (MMLL > 0)
Y
Secondary variables for ion-ion Redox CALL REDOX1
N
Precipitated mixed solid? (NBMIX > 0)
N
Y
Scan mixed solids Set K = 0
Next mixed solid K = k + 1
A precipitated mixed solid? (KMIX > 0)
N
Y
Select a secondary variable for mixed solid CALL MIXOL2
Last mixed solid? (K = NMIX)
N
Y


RETURN
N
Precipitated solids?
(NSOL > 0)
Y
Scan Solids
K = 0
Next solid
K = K + 1
A Precipitated
Solid?
(KPREC > 0)
N
Y
Secondary variable
for Precipitated Solid
CALL SOLID 2
Last Solid?
(K = NS)
N
Y
RETURN

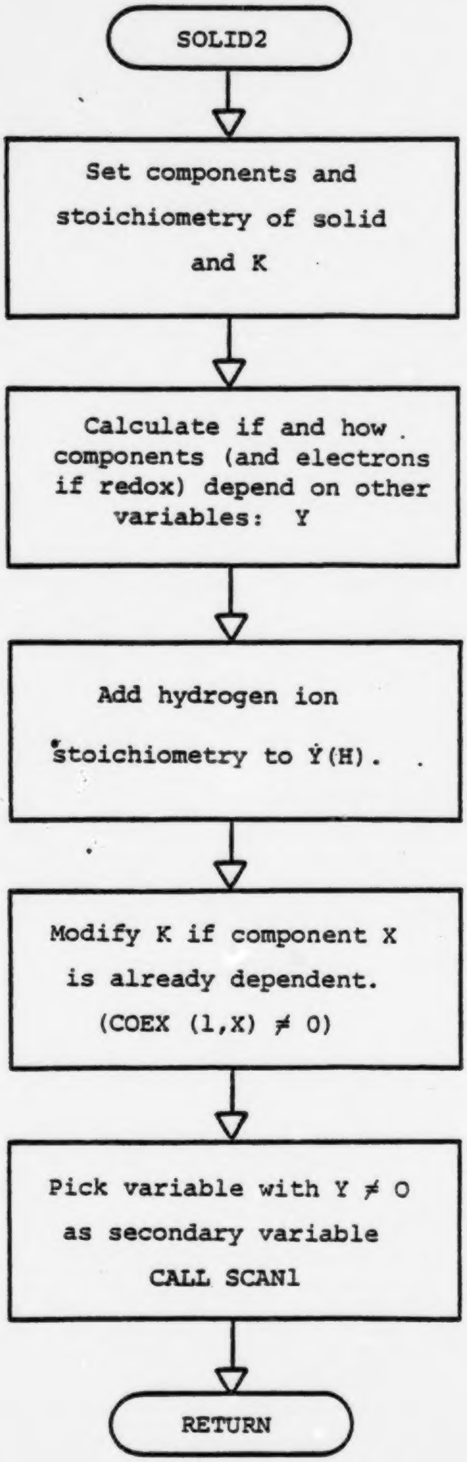
SKIP

SKIP is a simple-minded routine used by various input routines to perform dummy reads of the thermodynamic data base (i.e. skip lines). The parameter N specifies the number of lines to skip.


SKIP
N cards from the
Thermodynamic Data
File
RETURN

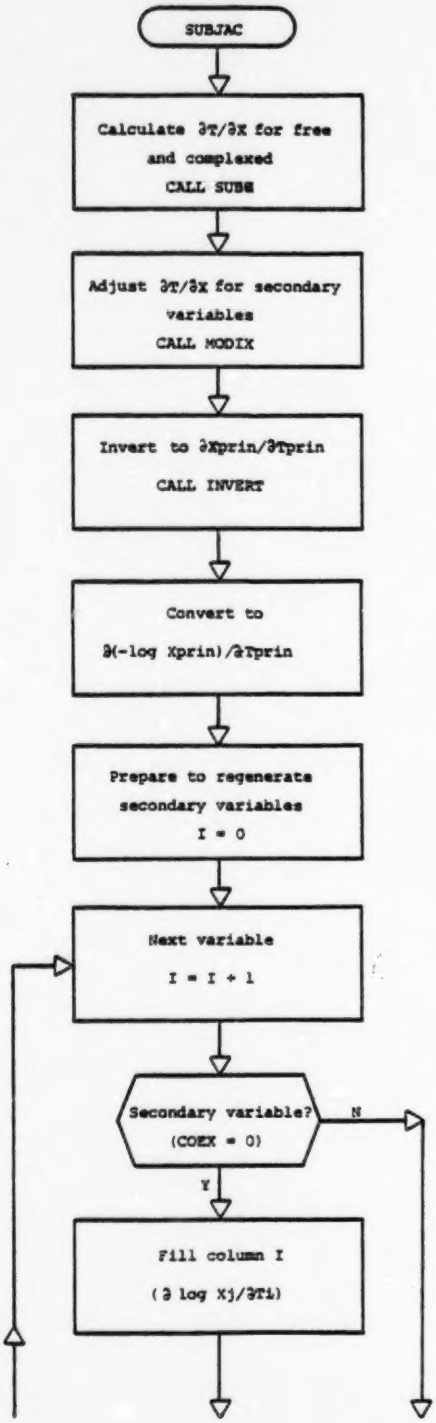
SOLID2

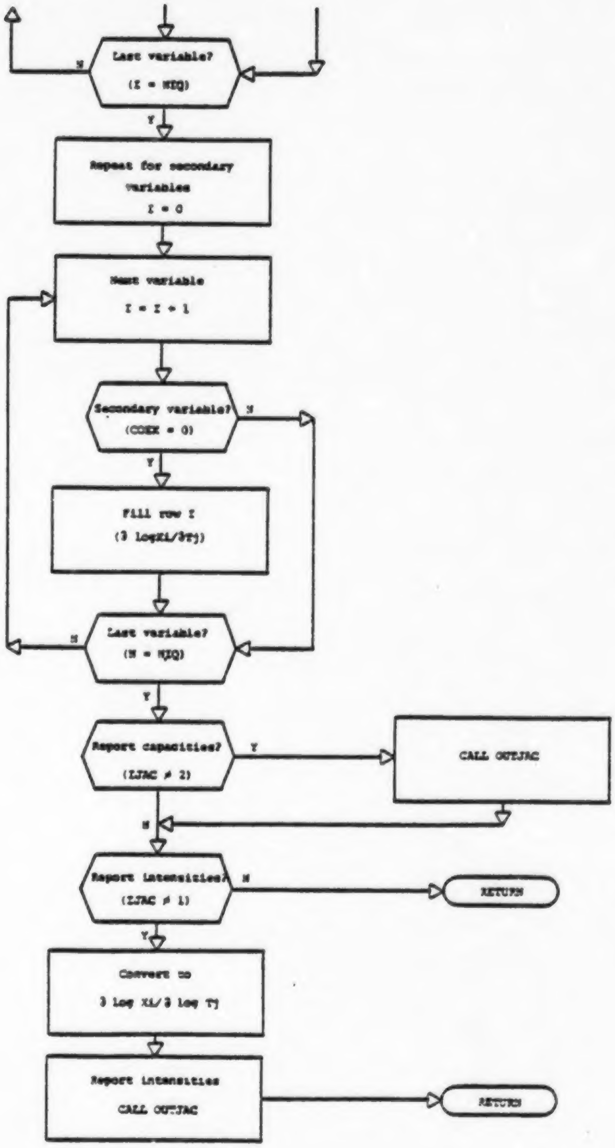
SOLID2 is called by SECVAR and CHKSOL to perform the first step in defining a secondary variable due to an imposed solid. The position of the solid is indicated by the parameter K. First, in a single loop, a list of candidates for the new secondary variable is selected; all principal variables upon which the metal or ligand of the solid depend become candidates and are flagged by the presence of a nonzero Y for the variable. If either of the components of the solid is already a secondary variable, the formation constant is modified. SCAN1 is called to complete the selection of the new secondary variable.


SOLID2
Set components and stoichiometry of solid and K
Calculate if and how components (and electrons if redox) depend on other variables: Y
Add hydrogen ion stoichiometry to Y(H).
Modify K if component X is already dependent. (COEX (1,X) ≠ 0)
Pick variable with Y ≠ 0 as secondary variable CALL SCAN1
RETURN

SUBJAC

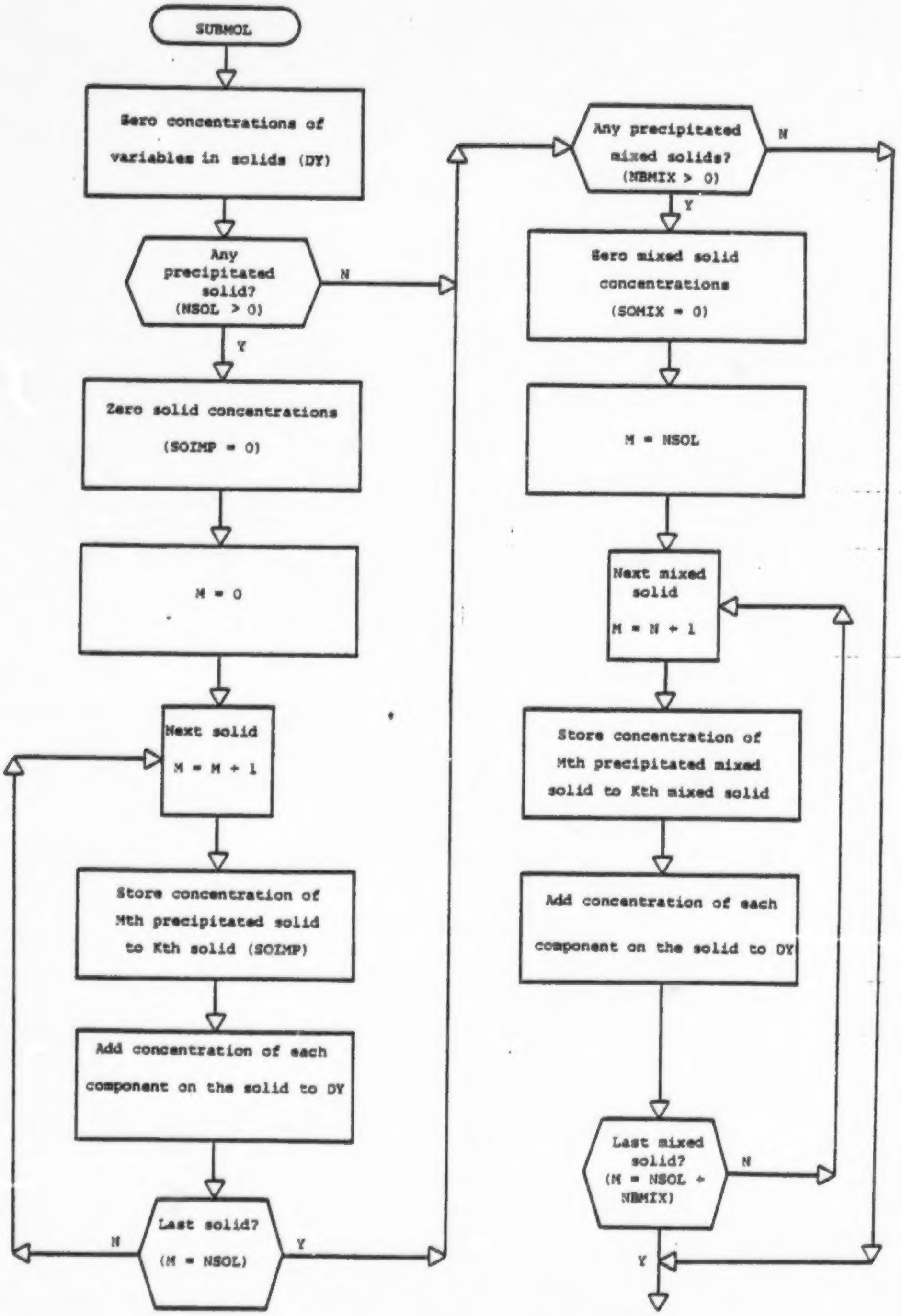
SUBJAC is called by SUB3 to calculate and report interaction capacities $\frac{\partial \log X_i}{\partial TOT_j}$ (IJAC=1), intensities $\frac{\partial \log X_i}{\partial \log TOT_j}$ (IJAC=2), or both (IJAC=3).

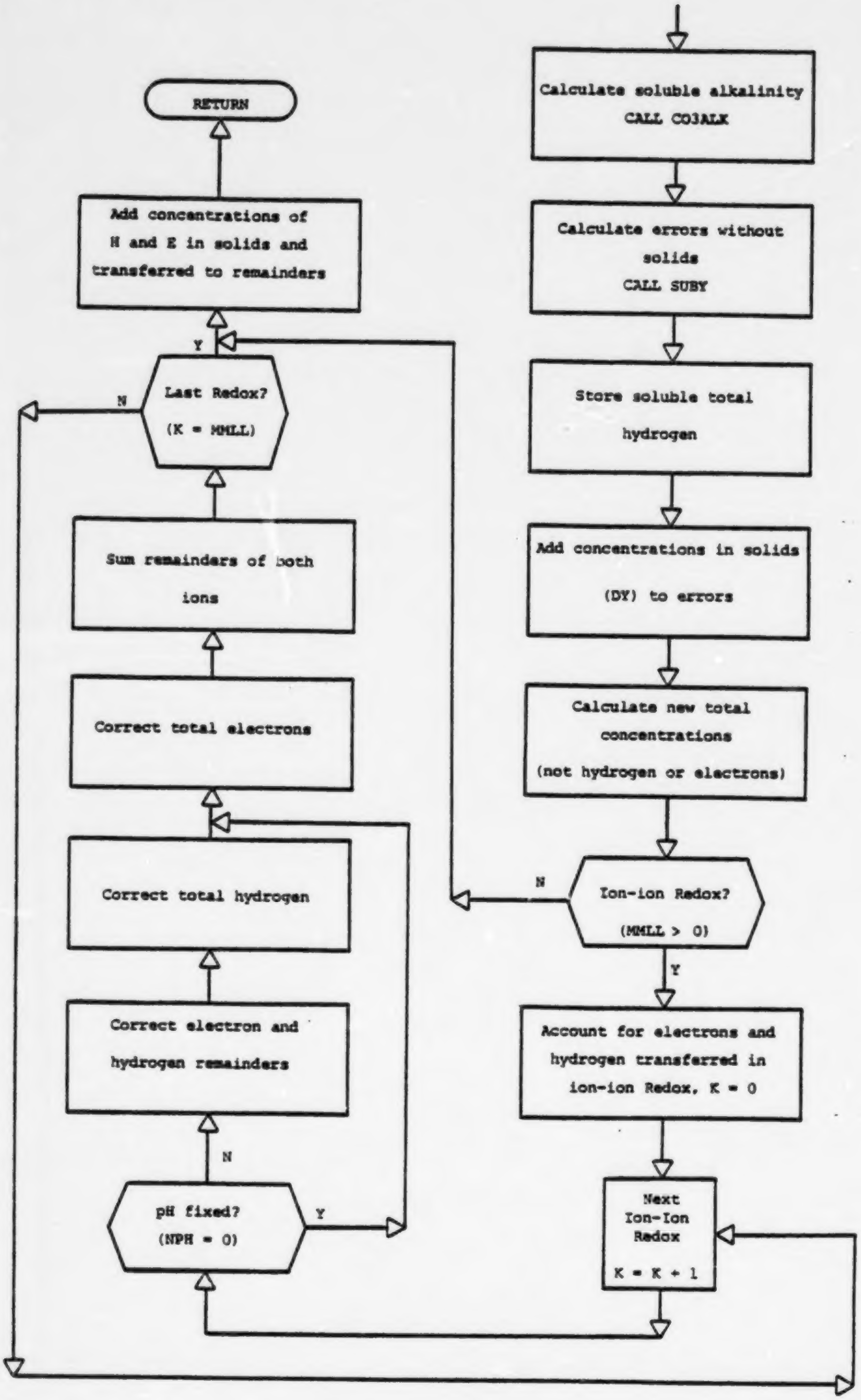

SUBJAC
Calculate ∂T/∂X for free and complexed CALL SUBG
Adjust ∂T/∂X for secondary variables CALL MODIX
Invert to ∂Xprin/∂Tprin CALL INVERT
Convert to ∂(-log Xprin)/∂Tprin
Prepare to regenerate secondary variables I = 0
Next variable I = I + 1
Secondary variable? (COEX = 0)
N
Y
Fill column I (∂ log Xj/∂Ti)


Last variable?
(I = NIQ)
N
Y
Repeat for secondary
variables
I = 0
Next variable
I = I + 1
Secondary variable?
(COEX = 0)
N
Y
Fill row I
(∂ logXi/∂Tj)
Last variable?
(N = NJQ)
N
Y
Report capacities?
(IJAC ≠ 2)
Y
CALL OUTJAC
N
Report intensities?
(IJAC ≠ 1)
N
RETURN
Y
Convert to
∂ log Xi/∂ log Tj
Report intensities
CALL OUTJAC
RETURN

SUBMOL

SUBMOL computes final remainders of variables in solids, including remainders for H^+ and electrons.


SUBMOL
Zero concentrations of variables in solids (DY)
Any precipitated solid? (NSOL > 0)
N
Y
Zero solid concentrations (SOIMP = 0)
M = 0
Next solid M = M + 1
Store concentration of Mth precipitated solid to Kth solid (SOIMP)
Add concentration of each component on the solid to DY
Last solid? (M = NSOL)
N
Y
Any precipitated mixed solids? (NBMIX > 0)
N
Y
Zero mixed solid concentrations (SOMIX = 0)
M = NSOL
Next mixed solid M = N + 1
Store concentration of Mth precipitated mixed solid to Kth mixed solid
Add concentration of each component on the solid to DY
Last mixed solid? (M = NSOL + NBMIX)
N
Y

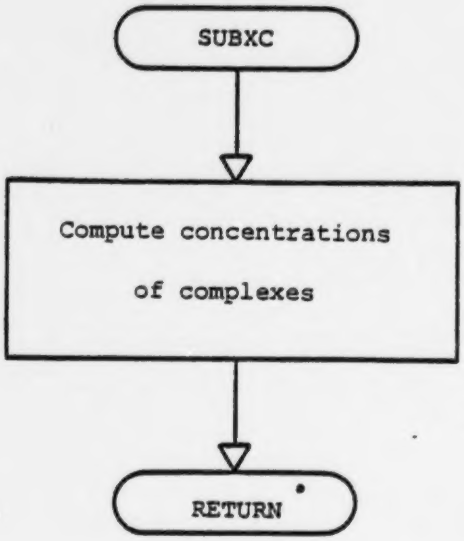

Calculate soluble alkalinity
CALL CO3ALK
Calculate errors without solids
CALL SUBY
Store soluble total hydrogen
Add concentrations in solids (DY) to errors
Calculate new total concentrations (not hydrogen or electrons)
Ion-ion Redox?
(MMLL > 0)
N
Y
Account for electrons and hydrogen transferred in ion-ion Redox, K = 0
Next Ion-Ion Redox
K = K + 1
pH fixed?
(NPH = 0)
Y
N
Correct electron and hydrogen remainders
Correct total hydrogen
Correct total electrons
Sum remainders of both ions
Last Redox?
(K = MMLL)
N
Y
Add concentrations of H and E in solids and transferred to remainders
RETURN

SUBXC

SUBXC is used by SUB3 to compute the concentration of all complexes by the formula:

[COMPLEX(i)] = KSTAB(i) * [[REACTANTS(i)] **STOICH(REACTANTS)]

The concentrations of all complexes are first computed as if they were "normal" complexes (i.e. a single metal and ligand plus either H^+ or OH^-). If there are complexes which contain an additional ion (i.e. mixed complexes), the concentrations of the affected complexes are modified. If any redox reactions are being considered, an additional loop adds the effect of electrons to the concentrations of redox complexes.

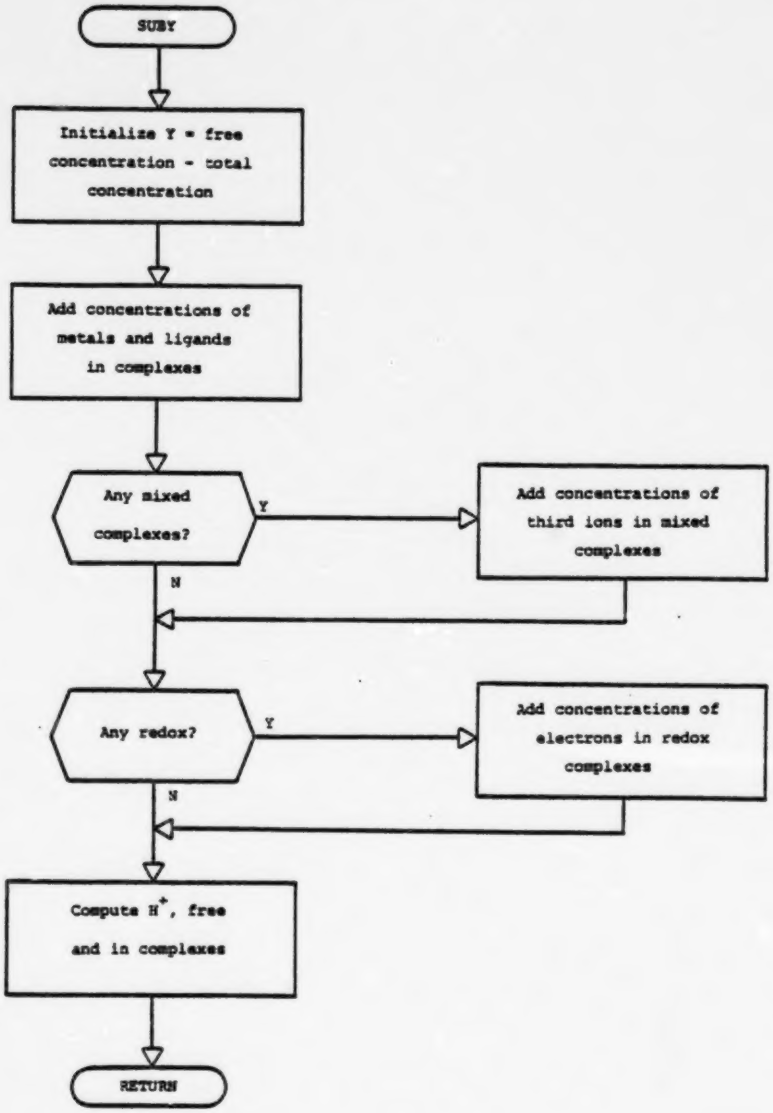

SUBXC
Compute concentrations of complexes
RETURN

SUBY

SUBY is used by SUB3 and SUBMOL to compute the "error vector", Y, prior to performing the test for convergence by subroutine CVTEST. The error Y(I) for each variable is initially defined as:

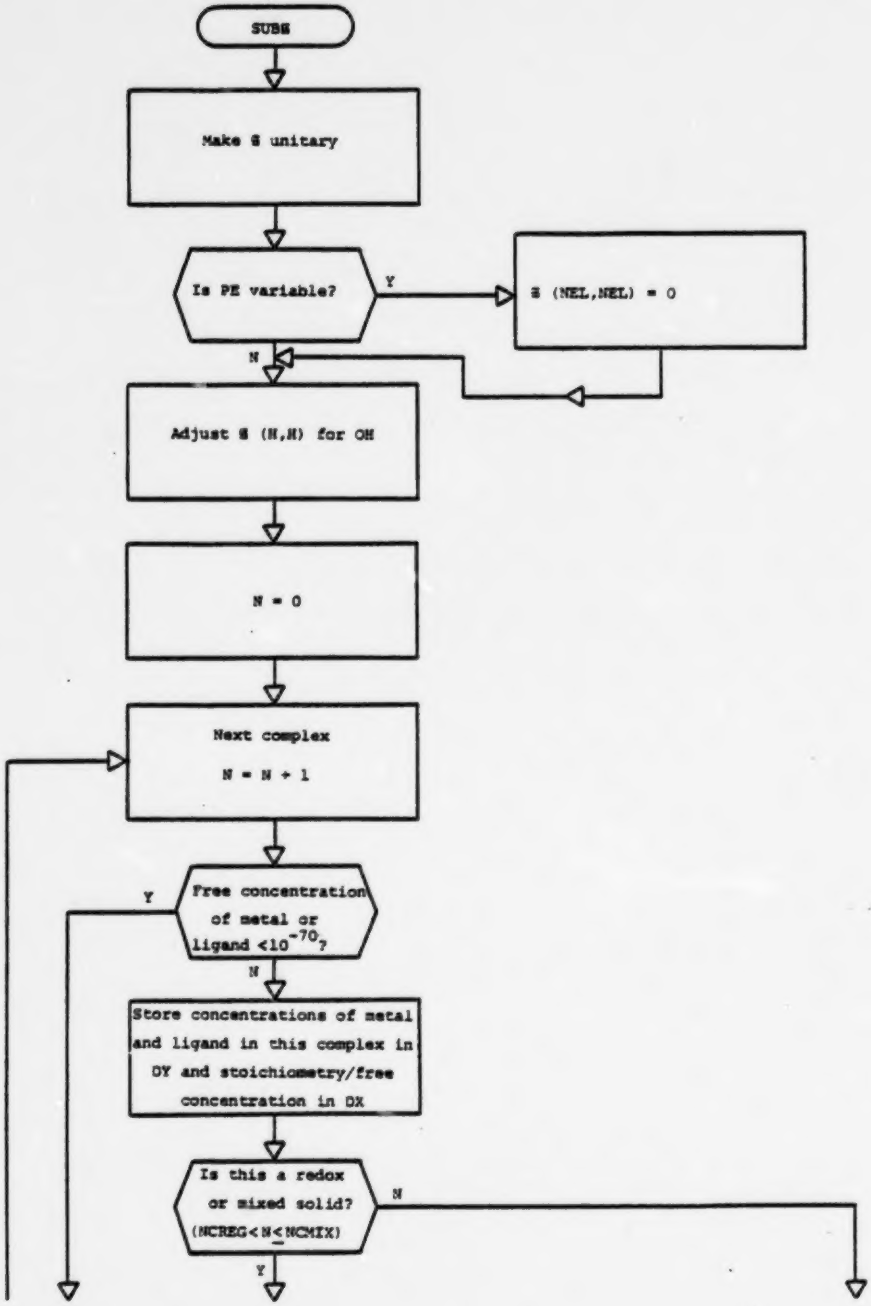
Y(I) = I(free) + I(complexed) - I(total)

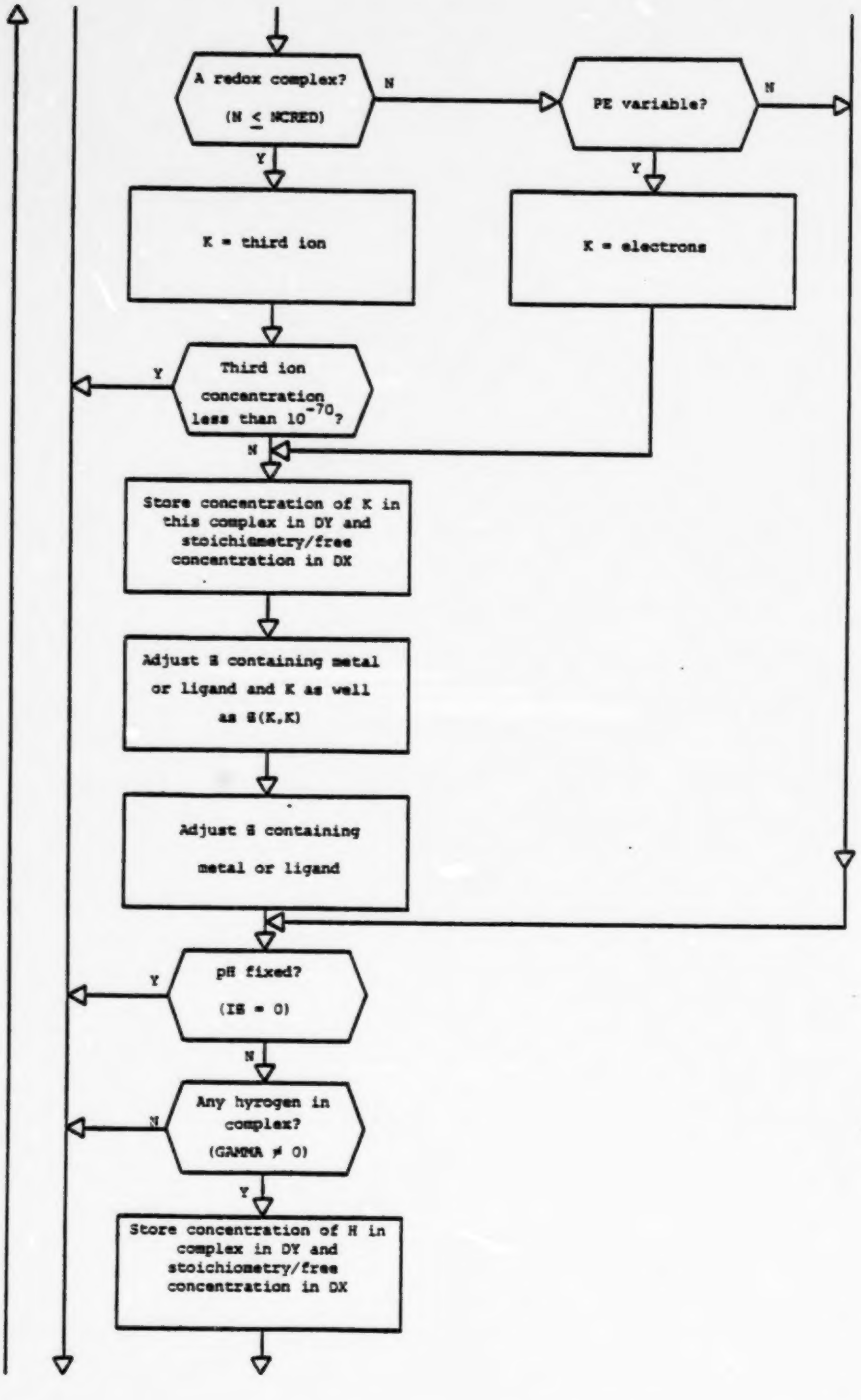
As usual, H^+ is a special case and is calculated separately. The calculation is begun by setting the error for each variable equal to the free concentration minus the total concentration of the ion (EXM(I) - TOTM(I)). The concentration of components involved in complexes then are added, keeping track of adjustments for H^+ as YH. The contribution of the extra element in mixed complexes (if any), are included and finally, the error for electrons is calculated if any redox reactions are included in the system.

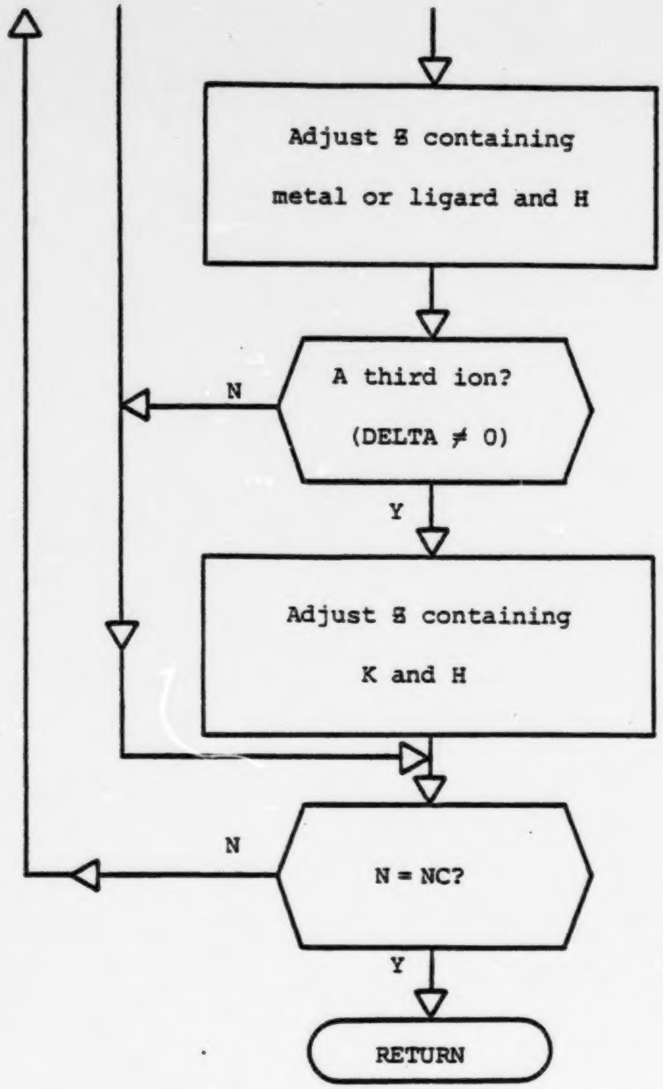

SUBY
Initialize Y = free concentration - total concentration
Add concentrations of metals and ligands in complexes
Any mixed complexes?
Y
Add concentrations of third ions in mixed complexes
N
Any redox?
Y
Add concentrations of electrons in redox complexes
N
Compute H+, free and in complexes
RETURN

SUBZ

Computes the Jacobian matrix (Z): the derivatives of total concentration with respect to free concentrations. SUBZ calculates derivatives for contributions of free and complexed ions. It is called with IZ which equals 0 if pH is fixed and 1 if pH is calculated.

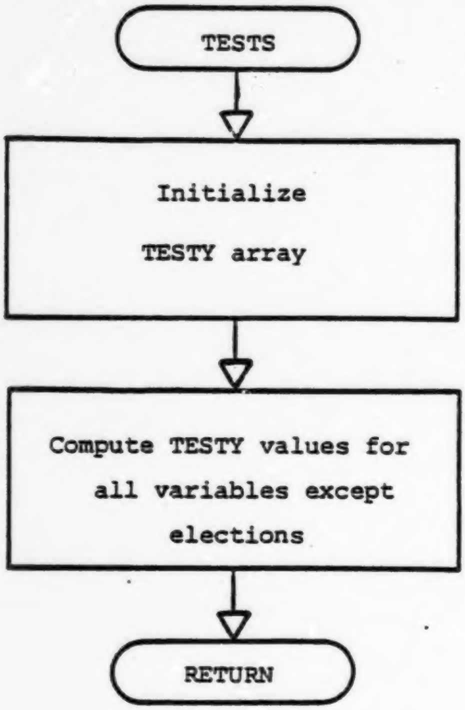

SUBS
Make S unitary
Is PE variable?
Y
S (NEL,NEL) = 0
N
Adjust S (H,H) for OH
N = 0
Next complex
N = N + 1
Y
Free concentration of metal or ligand <10^{-70}?
N
Store concentrations of metal and ligand in this complex in DY and stoichiometry/free concentration in DX
Is this a redox or mixed solid?
(NCREG< N ≤ NCMIX)
N
Y


A redox complex?
(N ≤ NCRED)
N
PE variable?
N
Y
Y
K = third ion
K = electrons
Third ion concentration less than 10^-70?
Y
N
Store concentration of K in this complex in DY and stoichiometry/free concentration in DX
Adjust S containing metal or ligand and K as well as S(K,K)
Adjust S containing metal or ligand
pH fixed?
(IS = 0)
Y
N
Any hyrogen in complex?
(GAMMA ≠ 0)
N
Y
Store concentration of H in complex in DY and stoichiometry/free concentration in DX


Adjust Ξ containing metal or ligard and H
A third ion? (DELTA ≠ 0)
N
Y
Adjust Ξ containing K and H
N = NC?
N
Y
RETURN

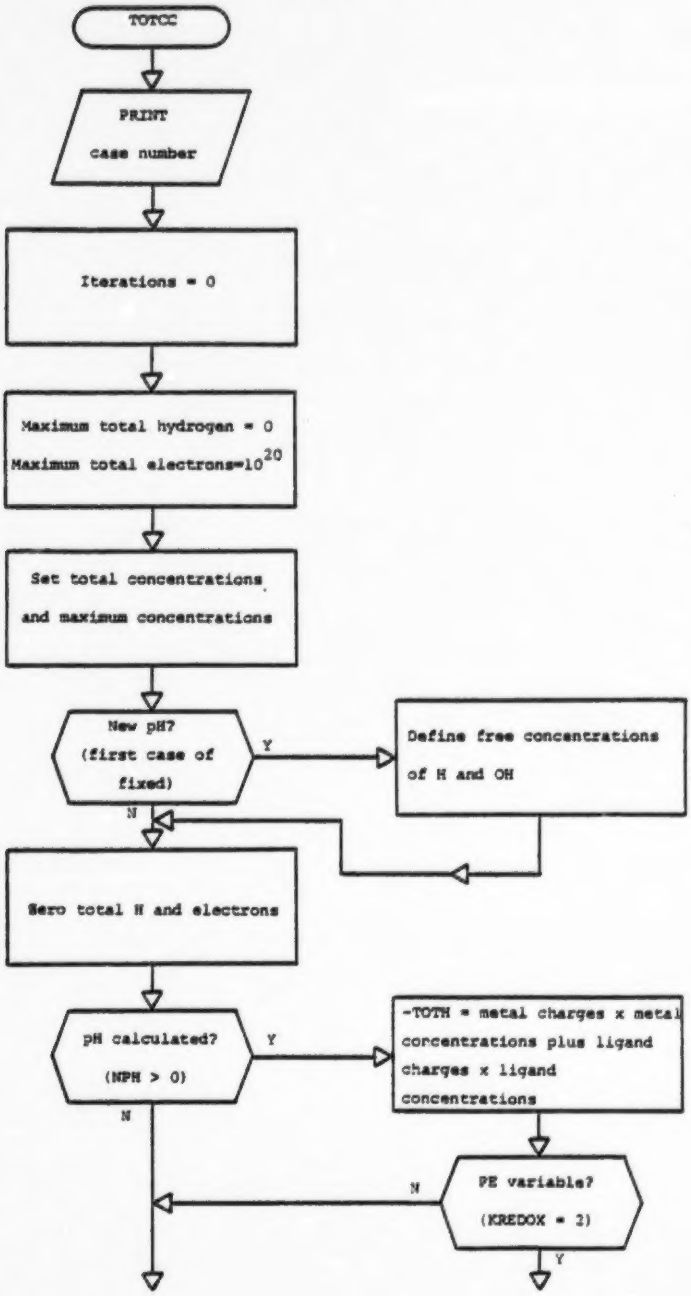
TESTS

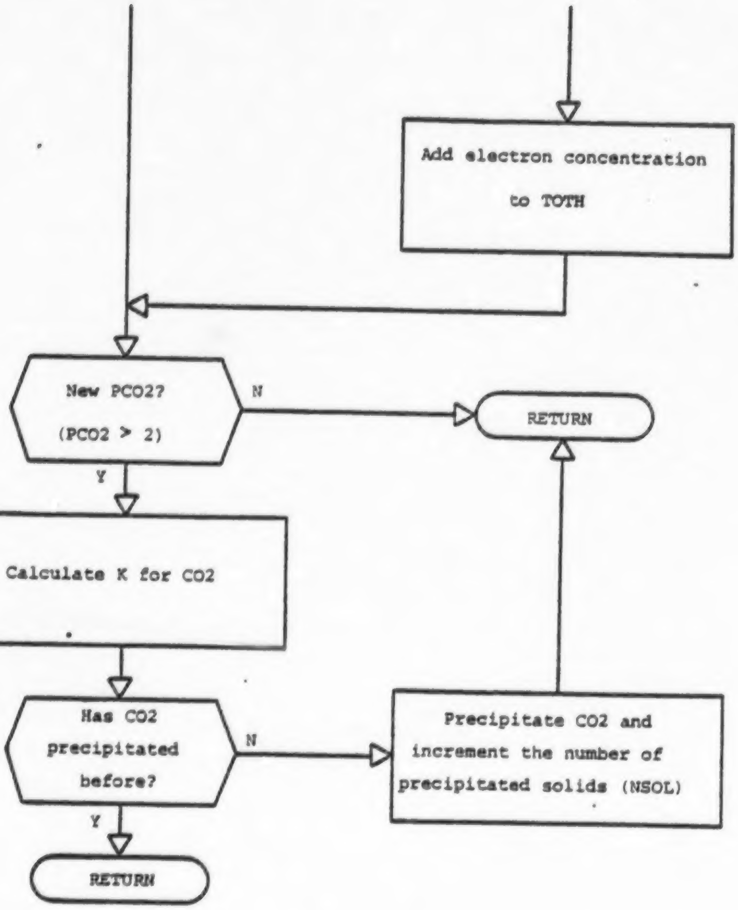
Subroutine TESTS is used by SUB3 to compute the TESTY array which will be used by CVTEST in performing the test for convergence. First the TESTY array is initialized to the corresponding values for total concentrations, TOTM. Then the TESTY values are computed for principal variables by setting them to the largest of their total concentrations or the total concentration of any secondary variable dependent upon them. Since the test for convergence will ignore secondary variables, they are given absurdly high values. Note that the TESTY value for hydrogen is computed differently; it has the largest value of all principal variables.


TESTS
Initialize
TESTY array
Compute TESTY values for
all variables except
elections
RETURN

TOTCC

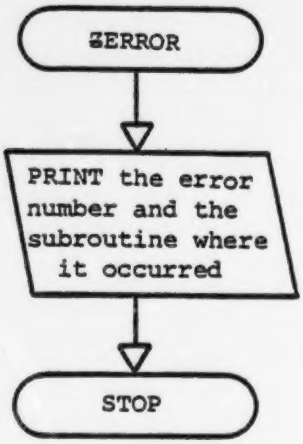
TOTCC is called by SUB3 at the beginning of each case to announce the new case and to perform initialization for the new case. After announcing the new case and initializing the iteration counter, the total concentrations for the case are imposed. TOTH and pH values are updated, and if CO2 is in the system, the PCO2 related values are updated and the "solid" CO2 is precipitated if this has not already been done.

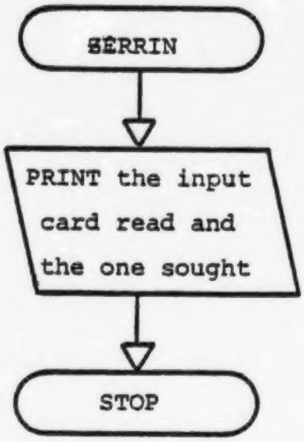

TOTCC
PRINT
case number
Iterations = 0
Maximum total hydrogen = 0
Maximum total electrons=10^20
Set total concentrations
and maximum concentrations
New pH?
(first case of
fixed)
Y
N
Define free concentrations
of H and OH
Zero total H and electrons
pH calculated?
(NPH > 0)
Y
N
-TOTH = metal charges x metal
concentrations plus ligand
charges x ligand
concentrations
PE variable?
(KREDOX = 2)
N
Y


Add electron concentration to TOTH
New PCO2?
(PCO2 > 2)
N
RETURN
Y
Calculate K for CO2
Has CO2 precipitated before?
N
Precipitate CO2 and increment the number of precipitated solids (NSOL)
Y
RETURN

ZERRIN and ZERROR

These two routines are used to inform the user of error conditions. ZERRIN reports input errors and ZERROR reports errors of program overflow.


$ERROR
PRINT the error number and the subroutine where it occurred
STOP


$ERRIN
PRINT the input card read and the one sought
STOP